Gaussian processes for sensitivity analysis, Bayesian inference, and uncertainty quantification in landslide research

Von der Fakultät für Georessourcen und Materialtechnik der
Rheinisch-Westfälischen Technischen Hochschule Aachen

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

genehmigte Dissertation

vorgelegt von

Hu Zhao, M. Eng.

aus Anhui, China

Berichtende: **Univ.-Prof. Dr. rer. nat. Klaus Reicherter**
Univ.-Prof. Dr. sc. Julia Kowalski

Tag der mündlichen Prüfung: 04.11.2021

Abstract

Landslides are common natural hazards occurring around the world. They pose an ongoing threat to lives, properties, and environment. Driven by the practical need to predict hazards of future landslides and design mitigation strategies, various physics-based landslide run-out models have been developed in the past decades. To achieve reliable and transparent simulation-based risk assessment and mitigation design, comprehensive understanding of the various uncertainties associated with these models is required. However, advanced statistical methods that are capable of properly addressing the uncertainties are often not applicable due to the computational bottleneck resulting from the relatively long run time of a single simulation and the large number of necessary simulations. To address the research gap, new methodologies are developed and studied in this thesis. They make up a unified framework that allows us to systematically, routinely, and efficiently investigate both forward and inverse problems resulting from the various uncertainties.

Chapter 1 introduces the background, frames the research gap, and motivates this study. Chapter 2 and 3 present theories of the two essential components of the unified framework, namely physics-based landslide run-out models and data-driven Gaussian process emulators. Chapter 4 presents a new methodology for efficient variance-based global sensitivity analyses of landslide run-out models. The methodology couples depth-averaged landslide run-out models, variance-based sensitivity analyses, robust multivariate Gaussian process emulation techniques, and an algorithm accounting for the emulator-uncertainty. Its feasibility and efficiency are validated by a case study based on the 2017 Bondo landslide event. The results show that it can recover common findings in the literature and provides further information on interactions between input variables along the full flow path. Chapter 5 presents a new methodology for efficient parameter calibration of landslide run-out models. It is developed by integrating depth-averaged landslide run-out models, Bayesian inference, Gaussian process emulation, and active learning. A case study using the new method is conducted based on the 2017 Bondo landslide event with synthetic observed data. The results show that the method is capable of correctly calibrating the rheological parameters and greatly improving the computational efficiency. Chapter 6 is devoted to uncertainty quantification of landslide run-out models. The focus is put on topographic uncertainty which is mostly overlooked in current practice. Two types of geostatistical methods are used to study the impact of topographic uncertainty on landslide run-out modeling based on the 2008 Yu Tung landslide event. It is found that topographic uncertainty significantly affects landslide run-out modeling, depending on how well the underlying flow path is represented. In addition, the close relation between the two geostatistical methods and Gaussian processes is revealed. Based on it, a new method that employs Karhunen–Loève expansion to reduce the dimensionality of topographic uncertainty is proposed. It has great potentials to make Gaussian process emulation

also applicable for high-dimensional topographic uncertainty and therefore allows us to treat topographic uncertainty within the unified framework. Chapter 7 provides concluding remarks and recommendations for future work.

Zusammenfassung

Erdrutsche sind eine weltweit häufig auftretende Naturgefahr. Sie stellen eine ständige Bedrohung für Leben, Eigentum und Umwelt dar. Aufgrund der Notwendigkeit, die Gefahren zukünftiger Erdrutsche vorherzusagen und Strategien zur Schadensbegrenzung zu entwickeln, wurden in den letzten Jahrzehnten verschiedene physikbasierte Auslaufmodelle für Rutschungen entwickelt. Um eine zuverlässige und transparente simulationsbasierte Risikobewertung und Schadeneindämmung zu erreichen, ist ein umfassendes Verständnis der verschiedenen mit diesen Modellen verbundenen Unsicherheiten erforderlich. Fortgeschrittene statistische Methoden, welche in der Lage sind, die Unsicherheiten angemessen zu berücksichtigen, sind jedoch aufgrund des Rechenengpasses, der sich aus der relativ langen Laufzeit einer einzelnen Simulation und der großen Anzahl notwendiger Simulationen ergibt, häufig nicht anwendbar. Um die hieraus resultierende Forschungslücke zu schließen, werden in dieser Arbeit neue Methoden entwickelt und untersucht. Sie bilden einen einheitlichen Rahmen, der es ermöglicht, sowohl Vorwärtsprobleme als auch inverse Probleme, die sich aus den verschiedenen Unsicherheiten ergeben, systematisch, routinemäßig und effizient zu untersuchen.

Kapitel 1 beschreibt den Hintergrund, erläutert die Forschungslücke und motiviert diese Arbeit. In den Kaiteln 2 und 3 werden die Theorien der beiden wesentlichen Komponenten des vereinheitlichten Frameworks vorgestellt, nämlich physikbasierte Auslaufmodelle für Rutschungen und datengesteuerte Gaußsche Prozessemulatoren. Kapitel 4 präsentiert eine neue Methodik für effiziente varianzbasierte globale Sensitivitätsanalysen von Rutschungsmodellen. Die Methodik koppelt tiefengemittelte Rutschungsmodelle, varianzbasierte Sensitivitätsanalysen, robuste multivariate Gaußsche Prozessemulationstechniken und einen Algorithmus, der die Emulatorunsicherheit berücksichtigt. Die Realisierbarkeit und Effizienz wird durch eine Fallstudie validiert, die auf dem Bondo-Erdrutschereignis 2017 basiert. Die Ergebnisse zeigen, dass die Methodik allgemeine Erkenntnisse aus der Literatur reproduzieren kann und weitere Informationen zu Wechselwirkungen zwischen Eingangsvariablen entlang des gesamten Rutschwegs liefert. In Kapitel 5 wird eine neue Methodik zur effizienten Parameterkalibrierung von Rutschungsmodellen vorgestellt. Die Methode wird durch die Integration von tiefengemittelten Rutschungsmodellen, bayessche Inferenz, Gaußscher Prozessemulation und aktivem Lernen entwickelt. Eine Fallstudie zur Anwendung der neuen Methode wird anhand des Bondo-Erdrutsches von 2017 mit synthetischen beobachteten Daten durchgeführt. Die Ergebnisse zeigen, dass die Methode in der Lage ist, die rheologischen Parameter korrekt zu kalibrieren und die Recheneffizienz erheblich zu verbessern. Kapitel 6 ist der Unsicherheitsquantifizierung von Rutschungsmodellen gewidmet. Der Fokus liegt auf der topografischen Unsicherheit, die in der aktuellen Praxis

meist übersehen wird. Es werden zwei Arten von geostatistischen Methoden verwendet, um die Auswirkungen der topografischen Unsicherheit auf die Modellierung des Erdrutschauslaufs basierend auf dem Yu Tung-Erdrutschereignis 2008 zu untersuchen. Es wird festgestellt, dass die topografische Unsicherheit die Modellierung des Erdrutschauslaufs erheblich beeinflusst, je nachdem, wie gut der zugrunde liegende Rutschweg dargestellt wird. Darüber hinaus wird der enge Zusammenhang zwischen den beiden geostatistischen Methoden und den Gaußschen Prozessen deutlich. Darauf aufbauend wird eine neue Methode vorgeschlagen, welche die Karhunen-Loeve-Expan-sion verwendet, um die Dimensionalität der topographischen Unsicherheit zu reduzieren. Diese Methode hat großes Potenzial, die Gaußsche Prozessemulation auch für hochdimensionale topografische Unsicherheiten anwendbar zu machen und erlaubt daher, topografische Unsicherheit in einem einheitlichen Rahmen zu behandeln. Kapitel 7 enthält abschließende Bemerkungen und Empfehlungen für zukünftige Arbeiten.

Acknowledgements

First and foremost, I want to thank my supervisors Professor Klaus Reicherter and Professor Julia Kowalski. Professor Reicherter has always been keen to help me whenever there was a need and provided me valuable comments and advice during our regular discussions. He always makes sure that I am on the right track. I am grateful to have the opportunity to work closely with Professor Kowalski. I sincerely appreciate her continued support and intelligent guidance throughout my PhD study. Without the many discussions we have had and her constructive comments, the accomplishment of this thesis and the publications would not have been possible. I also greatly appreciate her valuable advice on communication skills and personal development.

I would like to thank Professor Florian Amann for providing data for the Bondo case. I gratefully acknowledge his insightful comments and suggestions on the manuscript Emulator-based global sensitivity analysis for flow-like landslide run-out models. I would also like to thank Professor Martin Mergili for answering my questions about the software r.avaflow. My thanks also go to the Civil Engineering and Development Department of Hong Kong and DATA.GOV.HK for providing data for the Yu Tung case. Further thanks go to all the anonymous reviewers who peer-reviewed some individual chapters of this thesis.

I thank my graduate school the Aachen Institute for Advanced Study in Computational Engineering Science (AICES) and the Helmholtz School for Data Science in Life, Earth, and Energy (HDS-LEE) for all the wonderful courses and seminars, which helped me to broaden my knowledge and skill set. My sincere thanks also go to the AICES service team, the HDS-LEE coordinator Ramona Kloß, and all my colleagues for the friendly and supportive working environment. Special thanks go to my colleagues in the group of Professor Julia Kowalski – Alexander Zimmerman, Anil Yildiz, Ann-Kathrin Edrich, Anna Simson, Leonardo Boledi, Marc Boxberg, and Sascha Bodenburg – for all the help and memorable experience.

I express my gratitude to the China Scholarship Council for funding this work (grant number: 201706260262). I also thank the Statistical and Applied Mathematical Sciences Institute for the travel award to the Coupling Uncertain Geophysical Hazards Workshop, in which I learnt about emulation techniques. Further thanks go to the Helmholtz Information and Data Science Academy for funding my exchange to the High-Performance Computing Group at German Aerospace Center.

I am grateful to have my lovely flatmates Anu, Katja, Lorena, and Valentina, especially in hard times during a worldwide pandemic. We have had plenty of fun together. You guys make my life in Aachen much more colorful.

Finally, I would like to thank my parents for their continued support and unconditional love.

Contents

List of Figures

List of Tables

List of Abbreviations

CSS	**C**onditional **S**tochastic **S**imulation
DEM	**D**igital **E**levation **M**odel
GP	**G**aussian **P**rocess
MCMC	**M**arkov **C**hain **M**onte **C**arlo
RMSE	**R**oot **M**ean **S**quare **E**rror
USS	**U**nconditional **S**tochastic **S**imulation

Chapter 1

Introduction

1.1 Background and motivation

As one of the most common natural hazards, landslides occur frequently around the world and pose an ongoing threat to lives, properties, and environment. According to the 2005 World Bank report (Dilley et al., 2005), a total of about 3.7 million square kilometers land and 5 percent of total world population are subject to landslide hazards. A report on economic losses, poverty and disasters in 1998-2017 shows that 378 recorded landslides affected 4.8 million people and caused 18414 casualties and several billion US dollars of economic losses (Wallemacq, 2018). Froude and Petley (2018) reported that 4862 fatal non-seismic landslides in 2004-2016 killed in total 55997 people. The left panel of figure 1.1 shows a destructive earthquake-induced landslide in Las Colinas, Santa Tecla, 2001, which claimed 585 lives (García-Rodríguez and Malpica, 2010). Moreover, landslide occurrence and risk are expected to increase as a result of human activities, such as road construction and deforestation (Skilodimou et al., 2018; Li et al., 2020), and global warming that leads to increase of intensive precipitation events and permafrost melting (Stoffel et al., 2014; Shan et al., 2015; Gariano and Guzzetti, 2016).

Flow-like landslides that can move at high speeds and travel long distances have particularly high hazard potentials. The sketch shown in the right panel of figure 1.1 illustrates a typical flow-like landslide run-out process. Certain amount of flow mass is first initialized by for example an intensive precipitation event or an earthquake. The flow mass then flows over the topography driven by the gravity and finally stops and deposits due to friction. In order to assess the hazards and manage the risk, landslide run-out analyses are commonly conducted to obtain quantities of interest such as run-out distances, impact areas, impact pressures, deposit areas and volumes, flow heights and velocities, etc. Various models for run-out analyses have been developed by the geohazard community in past decades, including statistics-based empirical models and physics-based models.

Statistics-based empirical models are developed by applying statistical regression analyses to extensive historical data. The aim of this type method is to find correlations between interested quantities for hazard assessment (e.g. run-out distance, deposit area, etc.) and landslide types, release volumes, topographic characteristics,

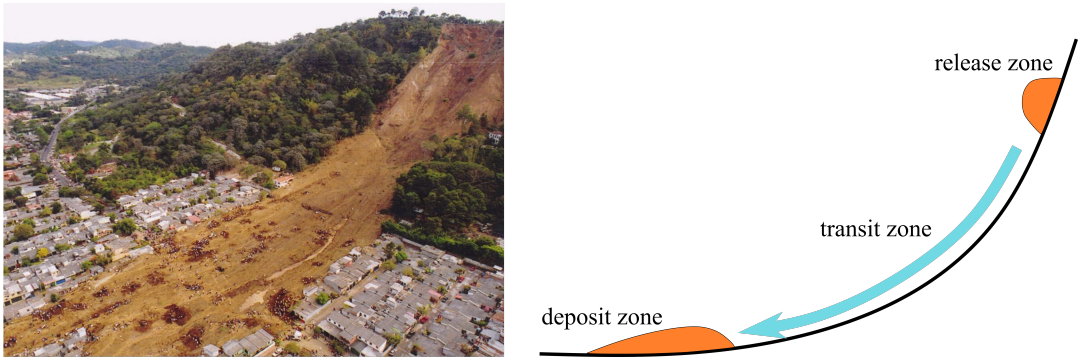

FIGURE 1.1: Left: An earthquake-induced landslide in Las Colinas, Santa Tecla, 2001 (García-Rodríguez and Malpica, 2010; reproduced under the CC Attribution 3.0 License). Right: Schematic illustration of a landslide run-out process.

etc. The correlation between the angle of reach and the release volume and the correlation between the deposit area and the release volume are two well-established examples (McDougall, 2017). This type of method is simple and practical, and is particularly appealing for rough overviews of landslide hazards at a regional scale (Pastor et al., 2012; Mergili et al., 2012). They however cannot provide detailed information on flow dynamics such as spatio-temporally resolved flow heights, flow velocities, impact pressures, etc. Such information is greatly useful for site-specific landslide hazard assessment and development of mitigation strategies.

In contrast, physics-based models allow assessing detailed flow dynamics. Due to the extremely complex nature of landslides, purely physics-based models are rare. Indeed, the majority of physics-based models used in both academia and practice are based on depth-averaged shallow flow equations (Pastor et al., 2012; McDougall, 2017; Rauter et al., 2018), also known as equivalent fluid models (Hungr, 1995; Aaron et al., 2019). More specifically, the heterogeneous flow material is treated as a hypothetical 'equivalent fluid' which is governed by simplified rheological relationships (Hungr, 2009). Such models are on the one hand physics-based, since they are usually derived based on soil and/or fluid mechanics. On the other hand, they are semi-empirical since the rheological parameters are more conceptual than physical (Fischer et al., 2015), which can only be obtained by back-analyzing past landslide events.

As systematically presented in section 2.3, these physics-based semi-empirical run-out models require various inputs, including the initial distribution of the flow mass, topographic data, and rheological parameters. All of them are subject to uncertainties. It is therefore desirable to develop a unified framework, which allows systematically and routinely accounting for the uncertainties in these run-out models. It is an important step towards reliable and efficient model-based landslide hazard mapping and risk management. Advanced techniques that are capable of addressing the uncertainties should be integrated with the run-out models in the unified framework, such as global sensitivity analyses (chapter 4), Bayesian inference (chapter 5), and Monte Carlo-type uncertainty quantification (chapter 6). However,

the usage of these advanced techniques is greatly hindered by the relatively long run time of the run-out models, since each of the techniques would require running a run-out model many (thousands of) times.

A straightforward approach to address the computational bottleneck is parallel computing and high performance computing. While greatly reducing the total computational time, it usually requires a large amount of computational resources. An alternative approach is surrogate modeling, also known as metamodeling or emulation. The essential idea of surrogate modeling is to approximate an expensive-to-run high-fidelity simulation model (also referred to as a simulator) by a cheap-to-run surrogate model (also known as a metamodel or an emulator). A wide variety of surrogate modeling methods have been developed in the past decades and been increasingly used by many scientific communities in recent years. Based on how the high-fidelity simulation model is treated, these methods can be broadly classified into two families, namely physics-based lower-fidelity modeling and data-driven response surface modeling (Razavi et al., 2012).

A physics-based lower-fidelity surrogate model is essentially a simplified (cheap-to-run) model of the original high-fidelity (expensive-to-run) simulation model. It can be obtained by for example projecting the governing equations of the original simulation model to a reduced dimension subspace (model order reduction methods), simplifying the underlying physics modeled by the original model, or simply reducing the numerical accuracy (Razavi et al., 2012; Asher et al., 2015). This type of surrogate modeling has great potentials but typically requires modifying codes of the original simulation model. It therefore does not fit to the purpose of developing a unified framework for various run-out models and corresponding solvers. In contrast, data-driven response surface modeling treats the high-fidelity simulation model as a black box and allows using existing run-out model solvers in a non-intrusive way. More specifically, a statistical surrogate model is built to mimic the relationships between inputs of the simulation model and interested outputs (responses) of the simulation model, solely based on input-output data from a limited number of simulation model runs. Gaussian process (GP) emulation, a well-known data-driven approach, is used in this thesis as a key building block of the unified framework due to its rich theoretical background and its ability to rigorously account for emulation-induced uncertainty.

In addition to emulating simulation models, GP regression has also been widely used to study spatial data such as the topographic data involved in this thesis (chapter 6). It is often known as kriging in the field of geostatistics. Detailed theories of Gaussian processes are presented in chapter 3.

1.2 Research objectives

The various uncertainties need to be comprehensively understood in order to achieve reliable model-based landslide hazard mapping and risk management. However,

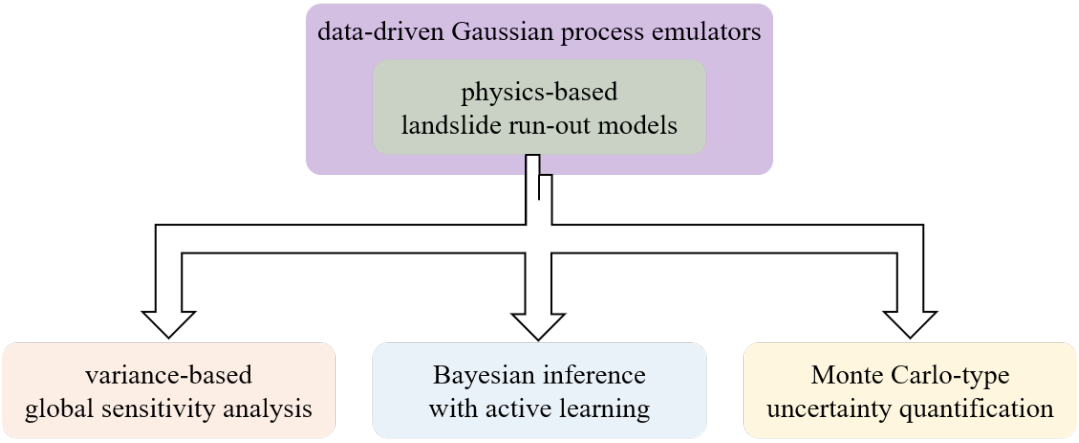

FIGURE 1.2: A schematic representation of the unified framework (a detailed version is shown in figure 7.1 in chapter 7).

they are often not properly addressed in current practice. For example, sensitivity analyses of landslide run-out models are commonly conducted using local one-at-a-time methods. These local methods cannot assess the interactions between different uncertain factors, which can be greatly helpful for us to better understand the models. Parameter calibration of landslide run-out models are usually based on deterministic trial-and-error methods, which overlook the uncertainty of calibrated parameters. The impact of topographic uncertainty on landslide run-out modeling has barely been studied in the literature.

To address the research gap, this thesis aims to develop a unified framework that allows us to systematically, routinely, and efficiently investigate various uncertainties associated with landslide run-out models. A schematic representation of the framework is shown in figure 1.2. It shows that the unified framework is built upon two essential components, namely physics-based landslide run-out models and data-driven Gaussian process emulators. Advanced statistical tools including the variance-based global sensitivity analysis, Bayesian inference with active learning, and Monte Carlo-type uncertainty quantification are integrated with the two core components in order to solve the forward problems (sensitivity analyses and uncertainty quantification) and inverse problems (parameter calibration) resulting from the various uncertain factors. More specific objectives of this thesis are as follows:

- Develop a new methodology that enables efficient global sensitivity analyses of landslide run-out models, by integrating landslide run-out modeling, GP emulation, and variance-based global sensitivity analyses. Investigate the model's sensitivity to uncertain inputs based on a test case.

- Develop a new methodology that allows efficient parameter calibration of landslide run-out models, by integrating landslide run-out modeling, GP emulation, Bayesian inference, and active learning. Study the impact of different observed data on calibration results based on a test case.

- Develop a new methodology that enables efficient uncertainty quantification of landslide run-out models, by integrating landslide run-out modeling, Monte Carlo-type simulation, GP emulation, and Karhunen–Loève expansion. Quantify the impact of topographic uncertainty on landslide run-out model outputs based on a case study.

1.3 Outline

Chapter 2 presents physics-based landslide run-out models. It begins with a literature review, followed by a detailed mathematical description of run-out models. Based on the mathematical equations, the model inputs and outputs are presented. Various uncertainties associated with the run-out models are discussed. This chapter ends with an emphasis on the research gap regarding landslide run-out modeling.

Chapter 3 presents theories of Gaussian processes. A literature review is first given, focusing on GP regression in the field of geostatistics and surrogate modeling. Next, the theory of GP regression is introduced based on a function with a scalar output, including how a GP posterior can be derived, how the hyperparameters can be learned, and how to sample from Gaussian processes. Techniques that extend GP regression to multi-output functions are discussed in the end.

Chapter 4 proposes a new methodology for efficient global sensitivity analyses of landslide run-out models. It is a slightly modified version of the published paper: Hu Zhao, Florian Amann, Julia Kowalski. Emulator-based global sensitivity analysis for flow-like landslide run-out models, *Landslides*, 18, 3299-3314, 2021. Modifications to the published paper have been made in this chapter to align it with the other chapters and to avoid redundancies.

Chapter 4 starts with a literature review and points out the deficiency of commonly used one-at-a-time sensitivity analysis methods. Then the theory of Sobol' sensitivity analysis and the new emulator-based methodology are presented, followed by the implementation. After that, a case study based on the 2017 Bondo landslide event is shown and results are discussed. Next, a short summary is given. Contribution of each co-author, main modifications to the published paper, and permission to reproduce the publication have been noted at the end of the chapter.

Chapter 5 proposes a new methodology for efficient parameter calibration of landslide run-out models. The state-of-the-art is first provided and the motivation is explained. Next, the theory of Bayesian inference and the new emulator-based Bayesian active learning methodology are presented in detail, followed by the implementation. Then a case study is conducted based on the 2017 Bondo landslide event with synthetic observed data, and the results are discussed. Last, a short summary is presented. A manuscript based on the content of this chapter is currently under review.

Chapter 6 is devoted to uncertainty quantification of landslide run-out models. The focus is put on topographic uncertainty since it is often neglected in current

practice and its high dimensionality poses a challenge to the emulation-based framework. This chapter is a modified version of the published paper: Hu Zhao, Julia Kowalski. Topographic uncertainty quantification for flow-like landslide models via stochastic simulations, *Natural Hazards and Earth System Sciences*, 20, 1441-1461, 2020. Modifications to the published paper have been made in this chapter to avoid redundancies, improve the coherence of this thesis as a whole, and include related work after the publication.

Chapter 6 first provides a literature review on topographic uncertainty studies. Next, two types of geostatistical methods are presented and used to study the impact of topographic uncertainty based on the 2008 Yu Tung landslide event. After that, a new methodology is presented. It uses Karhunen-Loève expansion to reduce the dimensionality of topographic uncertainty and therefore allows addressing topographic uncertainty in the unified emulation-based framework. Then, a short summary is presented. Contribution of each co-author, main modifications to the publication, and permission to reproduce the publication are noted in the end.

Chapter 7 summarizes the conclusions and gives recommendations for future work.

Chapter 2

Physics-based landslide run-out models

2.1 Literature review

Driven by the practical need to predict hazards of future landslides and design mitigation strategies, various physics-based run-out models have been developed in the past decades. Aiming at reviewing the state-of-art in landslide run-out models and assessing their commonality, two benchmarking exercises on landslide run-out analyses have been held in Hong Kong during the 2007 International Forum on Landslide Disaster Management and the 2018 second JTC1 (Joint Technical Committee on Natural Slopes and Landslides) Workshop on Triggering and Propagation of Rapid Flow-like Landslides respectively. Seventeen different models from thirteen teams worldwide have been compared during the fist benchmarking exercise and twelve different models from nine teams have been compared in the second benchmarking exercise. Detailed summaries of the benchmarking exercises have been published in the proceedings of the two events (Hungr et al., 2007; Pastor et al., 2018). Other review work, such as Pastor et al. (2012) and McDougall (2017), also provides an overview of the various run-out models. Table 2.1 presents a list of selected landslide run-out models to date.

The various physics-based run-out models can be broadly classified into lumped mass models, discontinuum models, and continuum models. In lumped mass models, the flow mass is assumed to be condensed to a single point. This assumption greatly simplifies the computation and allows analytical solutions for simple topographies (De Blasio, 2011). This type of model however does not account for internal deformation of the flow material and spatial distribution of flow dynamics (Chen and Lee, 2004; Mergili et al., 2012). Discontinuum models are based on discontinuum (granular) mechanics. They treat the flow mass as an assembly of particles and simulate the independent movement of each particle and their interactions. While being able to directly simulate three-dimensional flow behaviours and impact forces on structures (Teufelsbauer et al., 2011), discontinuum models are extremely computationally intensive, especially in the case of complex topographies and large scale landslide events.

TABLE 2.1: Selected run-out models (partly based on McDougall, 2017).

Model	Basic approach	Solution approach	Reference
lumped mass	lumped mass	analytical	De Blasio (2011)
PFC3D	discontinuum	distinct element	Poisel and Preh (2008)
Tochnog	differential	finite element	Crosta et al. (2003)
3dDMM	depth-averaged	particle-in-cell	Kwan and Sun (2007)
D-Claw	depth-averaged	finite volume	Iverson and George (2014)
DAMPM	depth-averaged	material point	Abe and Konagai (2016)
DAN3D	depth-averaged	smoothed particle hydrodynamics (SPH)	Hungr and McDougall (2009)
faSavage-HutterFOAM	depth-averaged	finite area	Rauter et al. (2018)
FLATModel	depth-averaged	finite volume	Medina et al. (2008)
FLO-2D	depth-averaged	finite difference	O'Brien et al. (1993)
GeoFlow-SPH	depth-averaged	SPH	Pastor et al. (2009)
GERRIS	depth-averaged	finite volume	Hergarten and Robl (2015)
MADFLOW	depth-averaged	finite element	Chen and Lee (2000)
Massflow-2D	depth-averaged	finite difference	Ouyang et al. (2013)
MassMov2D	depth-averaged	finite difference	Beguería et al. (2009)
r.avaflow	depth-averaged	finite difference	Mergili et al. (2017)
r.avalanche	depth-averaged	finite difference	Mergili et al. (2012)
RAMMS	depth-averaged	finite volume	Christen et al. (2010)
RASH3D	depth-averaged	finite volume	Pirulli and Mangeney (2008)
SHALTOP-2D	depth-averaged	finite volume	Mangeney-Castelnau et al. (2003)
TITAN2D	depth-averaged	finite volume	Pitman et al. (2003)
Xia-Liang	depth-averaged	finite volume	Xia and Liang (2018)

In contrast to discontinuum models, continuum models treat the flow mass as continuum material rather than as discrete particles. Their governing systems are derived from classical conservation laws of mass, momentum, and energy using continuum mechanics. Depending on whether the depth-averaged technique is employed, continuum models can be further divided into differential models and depth-averaged models (Hungr et al., 2007). Differential models, such as Tochnog (Crosta et al., 2003), focus on an element of the flow mass and are able of simulating fully three-dimensional flow behaviours. They are however computationally intensive and highly demanding on input data (Aaron and Hungr, 2016). Depth-averaged models focus on a column of the flow mass above the sliding surface by integrating the governing equations along the flow depth direction. By employing the shallow flow assumption (namely the flow depth is much smaller than the flow length), the depth-wise dimension is eliminated from the governing system. Depth-averaged models are therefore more computationally efficient than differential models.

Depth-averaged models have been proven to be capable of simulating the bulk behaviour of the flow mass. Validation of these models has been conducted in the past decades based on analytical benchmarks (Mangeney et al., 2000; Mangeney-Castelnau et al., 2003; Medina et al., 2008; Pastor et al., 2009; Xia and Liang, 2018), laboratory and field experiments (Savage and Hutter, 1989; Hungr, 1995; Denlinger and Iverson, 2001; Mangeney-Castelnau et al., 2003; Medina et al., 2008; Hungr and McDougall, 2009; George and Iverson, 2014; Xia and Liang, 2018), and historic

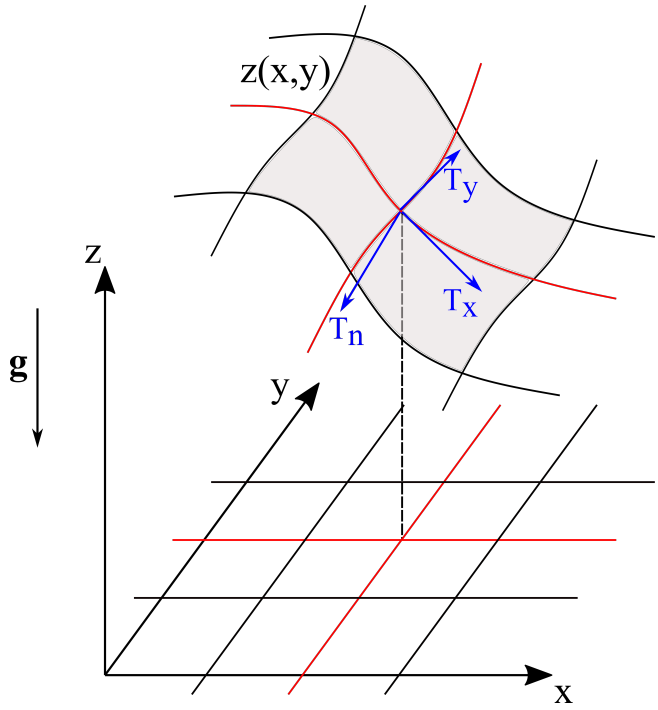

FIGURE 2.1: A sketch of a surface topography $z(x, y)$ in a Cartesian coordinate system {x,y,z} (based on Fischer et al., 2012). z axis points to the opposite direction of the gravitational acceleration, namely $\mathbf{g} = (0, 0, -g)^T$. The surface induces a local non-orthogonal coordinate system $\{T_x, T_y, T_n\}$, in which T_x and T_y are surface tangent directions and T_n denotes the surface normal direction.

events (Beguería et al., 2009; Christen et al., 2010; Mergili et al., 2012; Mergili et al., 2017; Rauter et al., 2018; Xia and Liang, 2018). Owing to their good compromise between computational efficiency and accuracy, depth-averaged models have gained popularity both in practice and in academia (Rauter et al., 2018). Indeed, the majority of run-out models to date are depth-averaged models, as shown in table 2.1. Some of them are available as commercial software, such as RAMMS (Christen et al., 2010) and FLO-2D (O'Brien et al., 1993). Many others are published as open source software such as D-Claw (Iverson and George, 2014), faSavageHutterFOAM (Rauter et al., 2018), GERRIS (Hergarten and Robl, 2015), r.avaflow (Mergili et al., 2017), TITAN2D (Pitman et al., 2003).

2.2 Mechanical and mathematical model

Pioneered by Savage and Hutter (1989), depth-averaged shallow flow type models have been greatly developed and used for simulating landslide run-out behaviours in recent decades. The governing system of the models is derived from the mass and momentum balance of the flow material. It can be expressed in a surface-induced coordinate system $\{T_x, T_y, T_n\}$ (see figure 2.1) as follows:

$$\partial_t h + \partial_x(hu_{T_x}) + \partial_y(hu_{T_y}) = \dot{Q}(x,y,t), \tag{2.1}$$

$$\partial_t(hu_{T_x}) + \partial_x\left(hu_{T_x}^2 + g_{T_n}k_{a/p}\frac{h^2}{2}\right) + \partial_y\left(hu_{T_x}u_{T_y}\right) = g_{T_x}h - S_{fT_x}, \tag{2.2}$$

$$\partial_t(hu_{T_y}) + \partial_x(hu_{T_x}u_{T_y}) + \partial_y\left(hu_{T_y}^2 + g_{T_n}k_{a/p}\frac{h^2}{2}\right) = g_{T_y}h - S_{fT_y}. \tag{2.3}$$

Equation 2.1 represents the mass balance. Equation 2.2 and equation 2.3 represent the momentum balance in the T_x and T_y direction respectively. In the equations, t denotes time; h represents flow height; u_{T_x} and u_{T_y} denote components of the depth-averaged surface tangent flow velocity $\mathbf{u} := (u_{T_x}, u_{T_y})^T$; g_{T_x},g_{T_y}, and g_{T_n} are components of the gravitational acceleration $\mathbf{g}$ along the surface tangent directions T_x and T_y, and surface normal direction T_n respectively.

The mass production source term $\dot{Q}(x,y,t)$ in equation 2.1 simulates the entrainment process (Christen et al., 2010). Various approaches have been proposed to simulate the entrainment process including both empirical and process-based methods (McDougall, 2017), such as Chen et al. (2006), Christen et al. (2010), Iverson (2012), and Frank et al. (2015). In this thesis, the entrainment process is not taken into account for simplicity. The developed unified framework and methods in following chapters can however be easily extended to account for it.

The active/passive earth pressure coefficient $k_{a/p}$ in equations 2.2-2.3 accounts for the elongation/compression of the flow material in the direction parallel to the surface. Savage and Hutter (1989) first introduced this concept into depth-averaged models and defined $k_{a/p}$ as

$$k_{a/p} = \frac{2}{\cos^2\phi}\left[1 \mp \sqrt{1 - (1+\tan^2\delta)\cos^2\phi}\right] - 1, \tag{2.4}$$

where ϕ and δ denote the internal friction angle and bed friction angle respectively. Equation 2.4 has been implemented in the open source solver r.avaflow (see table 2.1) which will be used in chapter 4 and chapter 5. Other forms of $k_{a/p}$ have also been used in run-out models, such as the one based on the Rankine's theory

$$k_{a/p} = \tan^2(45^\circ \mp \frac{\phi}{2}). \tag{2.5}$$

Equation 2.5 has been implemented in the commercial solver RAMMS (Christen et al., 2010) which will be used in chapter 6.

The friction terms S_{fT_x} and S_{fT_y} in equations 2.2-2.3 describe the basal rheology. A variety of flow resistance laws have been proposed and used in current practice including frictional, Voellmy, Bingham, Pouliquen, quadratic, Coulomb-viscous models, etc., see Naef et al. (2006), Hungr and McDougall (2009), and Pirulli and Mangeney (2008) for an overview. Among them, the Voellmy model has been widely

used especially in terms of flow-like landslides. It reads

$$S_{fT_x} = \frac{u_{T_x}}{\|\mathbf{u}\|}(\mu g_{T_n} h + \frac{g}{\xi}\|\mathbf{u}\|^2), \tag{2.6}$$

$$S_{fT_y} = \frac{u_{T_y}}{\|\mathbf{u}\|}(\mu g_{T_n} h + \frac{g}{\xi}\|\mathbf{u}\|^2), \tag{2.7}$$

where μ and ξ denote the dry-Coulomb friction coefficient and turbulent friction coefficient respectively. The Voellmy model will be used throughout this thesis. It should be noted that the proposed unified framework can be extended to other rheological models without loss of generality.

2.3 Model inputs and outputs

The governing system defined in equations 2.1-2.3 describes the time evolution of the state variables h and $\mathbf{u} := (u_{T_x}, u_{T_y})^T$. Various input data are required, including:

- *Initial distribution of the flow mass*. A landslide run-out analysis focuses on the post-failure motion of the flow mass. It means that the initial state of the state variables $h(x, y, t_0)$ and $\mathbf{u}(x, y, t_0)$ is needed. The initial velocity $\mathbf{u}(x, y, t_0)$ is typically zero and the initial distribution of the flow mass $h(x, y, t_0)$ determines the release area and volume. $h(x, y, t_0)$ can be given as a shape file (such as in RAMMS) or a raster map of release heights (such as in r.avaflow).

- *Topographic data*. The boundary topography on which the released mass flows is an important input since it determines the components of the gravitational acceleration g_{T_x}, g_{T_y}, and g_{T_n} in the local surface-induced coordinate system. It is commonly provided as a digital elevation model (DEM).

- *Rheological parameters*. The rheological parameters in the rheological models determine the flow resistance behaviour. In terms of the Voellmy rheology, they are the dry-Coulomb friction coefficient μ and turbulent friction coefficient ξ.

Given the above input data, the governing equations can be solved forward in time using various numerical methods, such as the finite element method, finite volume method, finite difference method, finite area method, smoothed particle hydrodynamics, material point method, etc., see table 2.1. The essential outputs are the flow height h and flow velocity $\mathbf{u}$ at each space-time grid point. By post-processing the spatio-temporally resolved h and $\mathbf{u}$, other interested quantities for landslide hazard assessment and mitigation can also be obtained, such as the run-out distance, angle of reach, apparent friction angle, impact area, deposit area and volume, pointwise impact pressure, etc. Detailed definitions of these derived outputs will be provided in the following chapters where these outputs are discussed.

2.4 Uncertain factors and research gap

When used for assessing and mitigating landslide hazards in practice, these depth-averaged shallow flow type run-out models involve a variety of uncertainties. The sources of uncertainties are as flows:

- *Uncertainty of the initial mass*. The release area and volume of a future landslide event is hardly predictable due to the complex geological pre-conditioning factors and a lack in the subsurface information. The estimation of the release area and volume still largely depends on expert experience. Probability density functions based on statistical properties of landslide inventories have also been proposed to approximate them (Quan Luna et al., 2013).

- *Uncertainty of topographic data*. Topographic data in the form of a digital elevation model are commonly based on remote sensing. Due to errors introduced during source data acquisition and data processing, topographic data are also subject to uncertainty.

- *Uncertainty of rheological parameters*. The rheological parameters used in the run-out models are more conceptual than physical as mentioned in the introduction. They can only be obtained by back-analyzing past events where field observation data are available. Using the back-analyzed values of rheological parameters for future landslide forecasting naturally involves uncertainty.

- *Uncertainty of observed data*. Observed data here refer to field data of past events which are used to calibrate the rheological parameters. This type of uncertainty is only concerned in the back-analyzing procedure but not in simulation-based landslide run-out forecasting.

The uncertainties have not been properly addressed in current practice. For example, the sensitivity of run-out models to the uncertain factors is commonly based on one-at-a-time sensitivity analyses which cannot explore the whole input space and account for interactions (see chapter 4). Rheological parameters are usually calibrated by trial-and-error calibration methods which ignore their uncertainty (see chapter 5). The impact of topographic uncertainty on landslide run-out modeling has barely been studied in the literature (see chapter 6). This may be partially due to the computational bottleneck resulting from the large number of simulation runs that are needed to properly address the uncertainties. To date, there is no unified framework that can systematically account for the various uncertainties.

Understanding the impacts of the various uncertainties and the ability of considering the uncertainties in a systematical and routine way are desirable due to the following reasons. First, understanding the impact of the uncertainties (like the model's sensitivity to the uncertain factors and their impact on simulation outputs) is greatly helpful from the viewpoint of model developers. Second, being able to systematically and routinely consider the uncertainties can significantly benefit the

usage of the run-out models from the viewpoint of practitioners. Lastly but most importantly, it is an important step towards reliable and transparent simulation-based hazard mapping, risk assessment, and mitigation strategy design.

I address the above-mentioned research gap in this thesis. The computational bottleneck is overcome by employing Gaussian process emulation. A unified framework that allows systematically, routinely, and efficiently accounting for the various uncertainties is developed. Forward problems like global sensitivity analyses and uncertainty quantification and inverse problems like Bayesian inference of rheological parameters can be solved under the unified framework. The open-source run-out solver r.avaflow (running on LINUX operating systems; Mergili et al., 2017) and the commercial run-out solver RAMMS (running on WINDOWS operating systems; Christen et al., 2010) are integrated in the unified framework and used for the case studies. The unified framework can be easily extended to other run-out models and solvers as listed in table 2.1 without loss of generality.

Chapter 3

Gaussian processes

3.1 Literature review

Gaussian processes are a generalization of multivariate Gaussian distributions to infinite dimensions. Regression with Gaussian processes has a long history. It dates back at least to Wiener (1949) who employed Gaussian processes for time series analyses. Later, GP regression has been developed and used in the context of geostatistics, surrogate modeling, machine learning, etc., see Rasmussen and Williams (2006) for a detailed overview.

In the geostatistics field, GP regression is known as kriging, named after the South African statistician and mining engineer Danie Krige due to his pioneering work (Krige, 1951). Kriging represents a family of generalized least-squares regression algorithms (Goovaerts, 1997). It was originally developed within the geostatistical community from the frequentist perspective (Sacks et al., 1989), see for example Matheron (1963). A Bayesian interpretation of kriging was later provided by for example Omre (1987) and Handcock and Stein (1993). This kind of approach has been widely used to interpolate spatial data (Cressie, 1993; Goovaerts, 1997; Stein, 1999) and generate realizations of random fields for uncertainty quantification (Holmes et al., 2000; Liu et al., 2019). In this thesis, it will be used to address topographic uncertainty in chapter 6.

In the context of surrogate modeling, GP regression is often referred to as GP emulation. Studies in this field are mainly driven by the fact that many computer models (referred to as simulators) can be computationally expensive to run, such as landslide run-out models described in chapter 2. The aim of GP emulation is to build cheap-to-run emulators based on input-output data of simulators. This type of approach has been developed from the Bayesian perspective since 1980s, see for example Sacks et al. (1989), Currin et al. (1991), and O'Hagan (2006). Many research efforts have been devoted to estimating the hyperparameters in GP emulators (see section 3.3) and extending the approach to multi-output simulators (see section 3.6). There also exists a body of literature regarding how to combine it with advanced statistical analysis methods such as global sensitivity analyses and Bayesian inference, which will be discussed in detail in chapter 4 and chapter 5 respectively.

Kriging and GP emulation are sometimes used interchangeably in the literature, such as Ba and Joseph (2012) and Maatouk and Bay (2017). To avoid confusion, I strictly distinguish the two terms. More specifically, kriging is used when it comes to geostatistics (namely topographic uncertainty in chapter 6), and GP emulation refers to build cheap-to-run emulators for simulators. The term Gaussian process regression is used to refer to both kriging and GP emulation.

3.2 Gaussian process regression

Regression represents a family of statistical methods that are designed to estimate unknown functions and make predictions. Traditional parametric regression methods, such as linear regression, specify a parametric form for the unknown function and place priors over the parameters of the form. In contrast, nonparametric regression methods do not predefine the form of the unknown function. Instead, they place a prior distribution directly over functions (Gershman and Blei, 2012). The level of complexity of the unknown function is directly determined by the training data used to update the prior distribution (Schulz et al., 2018).

As a well-known nonparametric Bayesian method, Gaussian process regression has been effectively used in many real-world applications (Mohammadi et al., 2019). In this method, the prior distribution over functions is determined by a Gaussian process. To understand a Gaussian process as a distribution over functions, one can informally view a function as an infinitely long vector, in which each entry specifies the function value at a specified input (Rasmussen and Williams, 2006). As a multivariate Gaussian distribution defines a distribution over a vector consisting of finite random variables, a Gaussian process which is a generalization of a multivariate Gaussian distribution to infinite dimensions defines a distribution over a vector consisting of infinite random variables. It can therefore be viewed as a distribution over functions.

3.2.1 Gaussian process prior

Let $f(\cdot)$ denote an unknown function that one wants to estimate. It defines a mapping from a p-dimensional input $\mathbf{x} = (x_1, \ldots, x_p)^T \in \mathcal{X} \subset \mathbb{R}^p$ to a scalar output $y \in \mathbb{R}$. In geostatistics, $\mathbf{x}$ usually consists of two- or three-dimensional spatial coordinates and y represents a quantity of interest that depends on the spatial coordinates such as the elevation error studied in chapter 6. In surrogate modeling of landslide run-out models, $\mathbf{x}$ could consist of uncertain variables such as the rheological parameters and release volume, and y could be an aggregated scalar output such as the impact area, see detailed examples in chapter 4. Here, an expensive-to-run landslide run-out model is viewed as an unknown function since the output corresponding to an input is unknown unless the run-out model has been run at that input.

From the Bayesian perspective, the prior belief of the unknown function $f(\cdot)$ is modeled by a Gaussian process, namely

$$f(\cdot) \sim \mathcal{GP}(m(\cdot), C(\cdot,\cdot)). \tag{3.1}$$

The Gaussian process is fully specified by its mean function $m(\cdot)$ and covariance function $C(\cdot,\cdot)$. The mean function is usually modeled by parametric regression

$$m(\mathbf{x}) = \mathbf{h}^T(\mathbf{x})\boldsymbol{\beta} = \sum_{t=1}^{q} h_t(\mathbf{x})\beta_t, \tag{3.2}$$

with q-dimensional vectors $\mathbf{h}(\mathbf{x}) = \left(h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_q(\mathbf{x})\right)^T$ and $\boldsymbol{\beta} = \left(\beta_1, \beta_2, \ldots, \beta_q\right)^T$ denoting the known basis functions and unknown regression parameters respectively. The basis functions are commonly chosen as constant $\mathbf{h}(\mathbf{x}) = 1$ or some prescribed polynomials like $\mathbf{h}(\mathbf{x}) = (1, x_1, \ldots, x_p)^T$, which correspond to ordinary kriging and universal kriging respectively (Ba and Joseph, 2012).

The covariance function $C(\cdot,\cdot)$ is also known as the covariogram (closely related to variogram or semivariogram, see section 6.2.2) in the geostatistics field (Cressie, 1993) and the kernel function in the machine learning field (Bishop, 2006). It can be either stationary or non-stationary. Stationary covariance functions depend only on the difference between inputs, namely $C(\mathbf{x}_i, \mathbf{x}_j) = C(\mathbf{x}_i - \mathbf{x}_j)$. As a special case of stationary covariance functions, isotropic covariance functions only depend on the distance between inputs, namely $C(\mathbf{x}_i, \mathbf{x}_j) = C(||\mathbf{x}_i - \mathbf{x}_j||)$. Rasmussen and Williams (2006) have discussed various isotropic covariance functions in detail, such as exponential covariance functions, rational quadratic covariance functions, the Matérn family, etc. New covariance functions can be constructed by modifying and/or combining existing covariance functions, see Rasmussen and Williams (2006) and Bishop (2006) for detailed techniques.

Isotropic covariance functions are often used in the geostatistics field such as Holmes et al. (2000), Temme et al. (2009), and Aziz et al. (2012). In the context of emulating complex simulators, the assumption of isotropy is often too strong (Gu et al., 2018). Instead, the covariance function is assumed to be stationary and typically has a separable form of

$$C\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sigma^2 c\left(\mathbf{x}_i, \mathbf{x}_j\right) = \sigma^2 \prod_{l=1}^{p} c_l\left(x_{il}, x_{jl}\right), \tag{3.3}$$

where $c\left(\mathbf{x}_i, \mathbf{x}_j\right)$ is the correlation function, and $c_l\left(x_{il}, x_{jl}\right)$ corresponds to the correlation between the l-th component of $\mathbf{x}_i$ and the counterpart of $\mathbf{x}_j$. Various correlation functions can be chosen for $c_l\left(x_{il}, x_{jl}\right)$. In what follows a specific correlation function from the Matérn family is chosen to illustrate the theory. It should be noted that it can be replaced by any other correlation function without loss of generality. The Matérn correlation function reads

$$c_l\left(x_{il}, x_{jl}\right) = \left(1 + \frac{\sqrt{5}||x_{il} - x_{jl}||}{\psi_l} + \frac{5||x_{il} - x_{jl}||^2}{3\psi_l^2}\right) \exp\left(-\frac{\sqrt{5}||x_{il} - x_{jl}||}{\psi_l}\right), \tag{3.4}$$

where ψ_l is the correlation length parameter in the l-th dimension. The unknowns

in the covariance function $C(\mathbf{x}_i, \mathbf{x}_j)$ are the variance σ^2 and the p correlation length parameters $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_p)^T$.

3.2.2 Gaussian process posterior

Equations 3.1-3.4 encode prior knowledge of the unknown function $f(\cdot)$. From the Bayesian viewpoint, this prior knowledge can be updated to a posterior based on a limited number of training data. Training data here refer to n^{tr} pairs of input-output data of the function $f(\cdot)$. In terms of topographic uncertainty, the training data consist of spatial coordinates of n^{tr} locations (inputs) and n^{tr} elevation errors measured at those locations (outputs). As for emulation of a landslide run-out model, the training data consist of n^{tr} model input configurations and corresponding model outputs by running the landslide run-out model at each input configuration.

Let $\mathbf{x}^{tr} := \{\mathbf{x}_i\}_{i=1}^{n^{tr}}$ and $\mathbf{y}^{tr} := \{f(\mathbf{x}_i)\}_{i=1}^{n^{tr}}$ denote the training inputs and outputs respectively. Since any finite number of random variables from a Gaussian process are jointly Gaussian distributed, the joint distribution of the n^{tr} outputs $\mathbf{y}^{tr}$ follow a n^{tr}-dimensional Gaussian distribution

$$p(\mathbf{y}^{tr}|\boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) \sim \mathcal{N}_{n^{tr}}\left(\mathbf{H}\boldsymbol{\beta}, \sigma^2\mathbf{R}\right), \tag{3.5}$$

where $\mathbf{H} = [\mathbf{h}(\mathbf{x}_1), \ldots, \mathbf{h}(\mathbf{x}_{n^{tr}})]^T$ is the $n^{tr} \times q$ basis design matrix and $\mathbf{R}$ is the $n^{tr} \times n^{tr}$ correlation matrix with (i, j) element $c(\mathbf{x}_i, \mathbf{x}_j)$. Equation 3.5 is known as the likelihood function.

Similarly, the joint distribution of the function output y^* at any untried input $\mathbf{x}^*$ and $\mathbf{y}^{tr}$ follows a $(n^{tr}+1)$-dimensional Gaussian distribution

$$p(y^*, \mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) \sim \mathcal{N}_{n^{tr}+1}\left(\begin{bmatrix} \mathbf{h}^T(\mathbf{x}^*) \\ \mathbf{H} \end{bmatrix} \boldsymbol{\beta}, \sigma^2 \begin{bmatrix} c(\mathbf{x}^*, \mathbf{x}^*) & \mathbf{r}^T(\mathbf{x}^*) \\ \mathbf{r}(\mathbf{x}^*) & \mathbf{R} \end{bmatrix}\right) \tag{3.6}$$

where $\mathbf{r}(\mathbf{x}^*) = (c(\mathbf{x}^*, \mathbf{x}_1), \ldots, c(\mathbf{x}^*, \mathbf{x}_{n^{tr}}))^T$. According to the property of the multivariate Gaussian distribution, the conditional distribution of y^* conditioned on $\mathbf{y}^{tr}$ is again a Gaussian distribution, namely

$$p(y^*|\mathbf{y}^{tr}, \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) \sim \mathcal{N}\left(m', \sigma^2 c'\right), \tag{3.7a}$$

$$m' = \mathbf{h}^T(\mathbf{x}^*)\boldsymbol{\beta} + \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\boldsymbol{\beta}), \tag{3.7b}$$

$$c' = c(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}^*). \tag{3.7c}$$

Equations 3.7a-3.7c can be easily extended to any dimensionality since the joint distribution of the function outputs at any number of untried inputs and $\mathbf{y}^{tr}$ also follows a multivariate Gaussian distribution. Equations 3.7a-3.7c therefore essentially define the posterior which is an updated Gaussian process, given the unknown regression parameters $\boldsymbol{\beta}$, variance σ^2, and correlation length parameters $\boldsymbol{\psi}$. The unknown parameters are also called hyperparameters.

For any new input $\mathbf{x}^*$, the output y^* can be almost instantaneously predicted by the mean m' (equation 3.7b). Moreover, the uncertainty associated with the prediction can be estimated by for example the variance $\sigma^2 c'$ or the 95% credible interval.

Depending on how the hyperparameters are treated, the posterior can have various forms different from equations 3.7a-3.7c which will be introduced in section 3.3.

3.3 Learning the hyperparameters

Learning the hyperparameters $\boldsymbol{\beta}$, σ^2 and $\boldsymbol{\psi}$ is a fundamental step in Gaussian process regression. The most straightforward approach is to find point estimates of the hyperparameters that maximize the likelihood defined in equation 3.5. The point estimates are then treated as true values of the hyperparameters and plugged into equation 3.7a. This is a typical non-Bayesian approach. In contrast, the hyperparameters can be marginalized out from equation 3.7a using a fully Bayesian approach. There are also approaches between non-Bayesian and fully Bayesian: some of the hyperparameters are marginalized out and some are based on point estimation. Details on above-mentioned approaches are presented as follows.

3.3.1 Maximum likelihood estimation

The likelihood function defined in equation 3.5 can be explicitly written as

$$p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) = \left(2\pi\sigma^2\right)^{-\frac{n^{tr}}{2}} |\mathbf{R}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2\sigma^2}\left(\mathbf{y}^{tr} - \mathbf{H}\boldsymbol{\beta}\right)^T \mathbf{R}^{-1}\left(\mathbf{y}^{tr} - \mathbf{H}\boldsymbol{\beta}\right)\right\}. \tag{3.8}$$

The point estimates $\hat{\boldsymbol{\beta}}_{ML}$, $\hat{\sigma}^2_{ML}$ and $\hat{\boldsymbol{\psi}}_{ML}$ (the subscript ML denotes maximum likelihood) that maximize equation 3.8 are the maximum likelihood estimates, namely

$$\hat{\boldsymbol{\beta}}_{ML}, \hat{\sigma}^2_{ML}, \hat{\boldsymbol{\psi}}_{ML} = \underset{\boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}}{\operatorname{argmax}} \ln\{p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi})\}. \tag{3.9}$$

Here, the log-likelihood is used to simplify the calculation. The point estimates of the regression parameters $\hat{\boldsymbol{\beta}}_{ML}$ and the variance $\hat{\sigma}^2_{ML}$ can be obtained in close forms by solving $\partial \ln\{p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi})\}/\partial\boldsymbol{\beta} = 0$ and $\partial \ln\{p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi})\}/\partial\sigma^2 = 0$. They are given by

$$\hat{\boldsymbol{\beta}}_{ML} = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{y}^{tr}, \tag{3.10a}$$

$$\hat{\sigma}^2_{ML} = \frac{1}{n^{tr}}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}_{ML})^T\mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}_{ML}). \tag{3.10b}$$

The correlation length parameters $\hat{\boldsymbol{\psi}}_{ML}$ do not have a closed form expression and can only be determined numerically, i.e. by maximizing equation 3.8 after plugging in $\hat{\boldsymbol{\beta}}_{ML}$ and $\hat{\sigma}^2_{ML}$ (Andrianakis and Challenor, 2009; Andrianakis and Challenor, 2011).

Given the maximum likelihood estimation $\hat{\boldsymbol{\beta}}_{ML}$, $\hat{\sigma}^2_{ML}$ and $\hat{\boldsymbol{\psi}}_{ML}$, the Gaussian process posterior defined in equations 3.7a-3.7c turns to

$$p(y^* \mid \mathbf{y}^{tr}, \hat{\boldsymbol{\beta}}_{ML}, \hat{\sigma}^2_{ML}, \hat{\boldsymbol{\psi}}_{ML}) \sim \mathcal{N}\left(m_{ML}, \hat{\sigma}^2_{ML} c_{ML}\right), \tag{3.11a}$$

$$m_{ML} = \mathbf{h}^T(\mathbf{x}^*)\hat{\boldsymbol{\beta}}_{ML} + \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}_{ML}), \tag{3.11b}$$

$$c_{ML} = c(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}^*). \tag{3.11c}$$

$\hat{\boldsymbol{\psi}}_{ML}$ appears in all terms that relate to the correlation function, including $c(\mathbf{x}^*, \mathbf{x}^*)$, $\mathbf{r}^T(\mathbf{x}^*)$, and $\mathbf{R}$. Equations 3.11a-3.11c indicate typical kriging equations.

3.3.2 Integrating out $\boldsymbol{\beta}$

To marginalize the regression parameters $\boldsymbol{\beta}$ from the GP posterior (equation 3.7a), a weak prior for $\boldsymbol{\beta}$ needs to be assumed. It has the form of $p(\boldsymbol{\beta}) \propto 1$, which can also be viewed as $p(\boldsymbol{\beta} \mid \sigma^2, \boldsymbol{\psi}) \propto 1$. The product of the weak prior and the likelihood (equation 3.8) leads to the posterior of $\boldsymbol{\beta}$, namely

$$p(\boldsymbol{\beta} \mid \mathbf{y}^{tr}, \sigma^2, \boldsymbol{\psi}) \propto p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) p(\boldsymbol{\beta} \mid \sigma^2, \boldsymbol{\psi}). \tag{3.12}$$

Then $\boldsymbol{\beta}$ can be integrated out from the GP posterior as follows:

$$p(y^* \mid \mathbf{y}^{tr}, \sigma^2, \boldsymbol{\psi}) = \int p(y^* \mid \mathbf{y}^{tr}, \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) p(\boldsymbol{\beta} \mid \mathbf{y}^{tr}, \sigma^2, \boldsymbol{\psi}) d\boldsymbol{\beta}. \tag{3.13}$$

The marginalization results in a new GP posterior $p(y^* \mid \mathbf{y}^{tr}, \sigma^2, \boldsymbol{\psi})$, conditioned on the training data, the variance σ^2, and the correlation length parameters $\boldsymbol{\psi}$.

The remaining hyperparameters σ^2 and $\boldsymbol{\psi}$ in the new GP posterior can be approximated by point estimates that maximize the following marginal likelihood (marginalizing $\boldsymbol{\beta}$)

$$p(\mathbf{y}^{tr} \mid \sigma^2, \boldsymbol{\psi}) = \int p(\mathbf{y}^{tr} \mid \boldsymbol{\beta}, \sigma^2, \boldsymbol{\psi}) p(\boldsymbol{\beta} \mid \sigma^2, \boldsymbol{\psi}) d\boldsymbol{\beta}. \tag{3.14}$$

Let $\hat{\sigma}^2_{MML}$ and $\hat{\boldsymbol{\psi}}_{MML}$ denote the point estimates that maximize equation 3.14. The subscript represents maximum marginal likelihood (marginalizing $\boldsymbol{\beta}$). $\hat{\sigma}^2_{MML}$ can be obtained in a closed form by solving $\partial \ln\{p(\mathbf{y}^{tr} \mid \sigma^2, \boldsymbol{\psi})\}/\partial\sigma^2 = 0$ and reads

$$\hat{\sigma}^2_{MML} = \frac{1}{n^{tr} - q} (\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}})^T \mathbf{R}^{-1} (\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}), \tag{3.15}$$

where $\hat{\boldsymbol{\beta}} = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{y}^{tr}$ has the same form as $\hat{\boldsymbol{\beta}}_{ML}$ (equation 3.10a). It should be noted that the regression parameters $\boldsymbol{\beta}$ have already been integrated out. $\hat{\boldsymbol{\beta}}$ is merely used to simplify the equations by convention. $\hat{\boldsymbol{\psi}}_{MML}$ is not analytically trackable and requires numerical approximation.

The new GP posterior is completed by plugging $\hat{\sigma}^2_{MML}$ and $\hat{\boldsymbol{\psi}}_{MML}$ into equation 3.13, and reads

$$p(y^* \mid \mathbf{y}^{tr}, \hat{\sigma}^2_{MML}, \hat{\boldsymbol{\psi}}_{MML}) \sim \mathcal{N}\left(m_{MML}, \hat{\sigma}^2_{MML} c_{MML}\right), \tag{3.16a}$$

$$m_{MML} = \mathbf{h}^T(\mathbf{x}^*)\hat{\boldsymbol{\beta}} + \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}), \tag{3.16b}$$

$$\begin{aligned} c_{MML} = c(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}^*) + \left(\mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{H} - \mathbf{h}^T(\mathbf{x}^*)\right) \\ \times (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1} \left(\mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{H} - \mathbf{h}^T(\mathbf{x}^*)\right)^T. \end{aligned} \tag{3.16c}$$

$\hat{\boldsymbol{\psi}}_{MML}$ is included in all terms that relate to the correlation function, including $c(\mathbf{x}^*, \mathbf{x}^*)$, $\mathbf{r}^T(\mathbf{x}^*)$, and $\mathbf{R}$.

3.3.3 Integrating out $\boldsymbol{\beta}$ and σ^2

Similar as the marginalization of the regression parameters $\boldsymbol{\beta}$, the variance σ^2 can be further integrated out from equation 3.13 by assuming a weak prior for σ^2. The weak prior is given by $p(\sigma^2) \propto (\sigma^2)^{-1}$ and can be treated as $p(\sigma^2 \mid \boldsymbol{\psi}) \propto (\sigma^2)^{-1}$. The posterior of σ^2 is obtained by using the Bayes' theorem, namely

$$p(\sigma^2 \mid \mathbf{y}^{tr}, \boldsymbol{\psi}) \propto p(\mathbf{y}^{tr} \mid \sigma^2, \boldsymbol{\psi}) p(\sigma^2 \mid \boldsymbol{\psi}), \tag{3.17}$$

where the first term on the right hand side is given by equation 3.14. Then σ^2 can be marginalized from equation 3.13 by

$$p(y^* \mid \mathbf{y}^{tr}, \boldsymbol{\psi}) = \int p(y^* \mid \mathbf{y}^{tr}, \sigma^2, \boldsymbol{\psi}) p(\sigma^2 \mid \mathbf{y}^{tr}, \boldsymbol{\psi}) d\sigma^2. \tag{3.18}$$

The marginalization of σ^2 leads to a new GP posterior $p(y^* \mid \mathbf{y}^{tr}, \boldsymbol{\psi})$, which is no longer a Gaussian process but a Student's-t process.

The left hyperparameters $\boldsymbol{\psi}$ in the Student's-t process can be approximated by maximum marginal likelihood estimates, namely point estimates that maximize the following marginal likelihood (marginalizing both $\boldsymbol{\beta}$ and σ^2)

$$p(\mathbf{y}^{tr} \mid \boldsymbol{\psi}) = \int p(\mathbf{y}^{tr} \mid \sigma^2, \boldsymbol{\psi}) p(\sigma^2 \mid \boldsymbol{\psi}) d\sigma^2. \tag{3.19}$$

Alternatively, $\boldsymbol{\psi}$ can be approximated by maximum marginal posterior estimates, namely point estimates that maximize the posterior of $\boldsymbol{\psi}$ given by

$$p(\boldsymbol{\psi} \mid \mathbf{y}^{tr}) \propto p(\mathbf{y}^{tr} \mid \boldsymbol{\psi}) p(\boldsymbol{\psi}). \tag{3.20}$$

Here, $p(\boldsymbol{\psi})$ is the chosen prior for $\boldsymbol{\psi}$. The maximum marginal posterior estimates are argued to be more robust than the maximum marginal likelihood estimates (Gu et al., 2018).

Let $\hat{\boldsymbol{\psi}}$ denote the point estimates that either maximize equation 3.19 or equation 3.20. The Student's-t process is then completed by plugging $\hat{\boldsymbol{\psi}}$ into equation 3.18 and reads

$$p(y^* \mid \mathbf{y}^{tr}, \hat{\boldsymbol{\psi}}) \sim \mathcal{S}t(m'', \hat{\sigma}^2 c'', n^{tr} - q), \tag{3.21a}$$

$$m'' = \mathbf{h}^T(\mathbf{x}^*)\hat{\boldsymbol{\beta}} + \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}), \tag{3.21b}$$

$$\hat{\sigma}^2 = \frac{1}{n^{tr} - q}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}})^T \mathbf{R}^{-1}(\mathbf{y}^{tr} - \mathbf{H}\hat{\boldsymbol{\beta}}), \tag{3.21c}$$

$$\begin{aligned} c'' = c(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{r}(\mathbf{x}^*) + \left(\mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{H} - \mathbf{h}^T(\mathbf{x}^*)\right) \\ \times (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\left(\mathbf{r}^T(\mathbf{x}^*)\mathbf{R}^{-1}\mathbf{H} - \mathbf{h}^T(\mathbf{x}^*)\right)^T. \end{aligned} \tag{3.21d}$$

It should be noted that $\boldsymbol{\beta}$ and σ^2 have been integrated out. $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ are only used to simplify the equations. Also, equations 3.21a-3.21d show the univariate Student's-t distribution rather than the Student's-t process defined in infinite dimensions. The equations can however be easily extended to any dimension. In addition, all terms that relate to the correlation function, including $c(\mathbf{x}^*, \mathbf{x}^*)$, $\mathbf{r}^T(\mathbf{x}^*)$, and $\mathbf{R}$, depend on $\hat{\boldsymbol{\psi}}$. Therefore m'' (equation 3.21b) and c'' (equation 3.21d) are different from m_{MML} (equation 3.16b) and c_{MML} (equation 3.16c) even though they have the same forms.

3.3.4 Integrating out $\boldsymbol{\beta}$, σ^2 and $\boldsymbol{\psi}$

To integrate out $\boldsymbol{\psi}$, one needs to solve the following integral

$$p(y^* \mid \mathbf{y}^{tr}) = \int p(y^* \mid \mathbf{y}^{tr}, \boldsymbol{\psi}) p(\boldsymbol{\psi} \mid \mathbf{y}^{tr}) d\boldsymbol{\psi}. \tag{3.22}$$

Unlike the marginalization of $\boldsymbol{\beta}$ (equation 3.13) and σ^2 (equation 3.18), the integral in equation 3.22 is not analytically trackable and can only be numerically approximated by for example Markov chain Monte Carlo methods. The resulted GP posterior $p(y^* \mid \mathbf{y}^{tr})$ does not have a closed form.

Details on this type of approach, namely numerically marginalizing $\boldsymbol{\psi}$, can be found in Andrianakis and Challenor (2009) and Garbuno-Inigo et al. (2016) and the references contained therein.

3.3.5 Short summary

Ideally, the fully Bayesian approach in which all the hyperparameters are marginalized (section 3.3.4) is preferred since it can fully take hyperparameter uncertainty into account. Numerically marginalizing the correlation length parameters $\boldsymbol{\psi}$ is however computationally difficult. It is therefore common in practice to treat the regression parameters $\boldsymbol{\beta}$ and the variance σ^2 from the Bayesian viewpoint and approximate $\boldsymbol{\psi}$ by point estimates in a non-Bayesian way (section 3.3.3). This type of approach will be used in chapter 4 where GP emulation is used to directly emulate input-output relations of landslide run-out models.

The maximum likelihood estimation (section 3.3.1) or marginal likelihood estimation (marginalize $\boldsymbol{\beta}$ and approximate σ^2 and $\boldsymbol{\psi}$ by point estimates; see section 3.3.2) leads to a GP posterior that is still a Gaussian process. In contrast, the marginal likelihood estimation (marginalize $\boldsymbol{\beta}$ and σ^2, and approximate $\boldsymbol{\psi}$ by point estimates; see section 3.3.3) results in a GP posterior that is no longer a Gaussian process but a Student's-t process. Due to good properties of Gaussian processes, the maximum likelihood estimation and marginal likelihood estimation (only marginalize $\boldsymbol{\beta}$) are also commonly used in practice, especially in the literature regarding Bayesian active learning for parameter calibration (see chapter 5).

3.4 Sampling

A Gaussian process, either the prior defined in equation 3.1 or the posterior essentially given by equations 3.11a-3.11c or equations 3.16a-3.16c, defines a distribution over functions. Once its mean function $m(\cdot)$ and covariance function $C(\cdot, \cdot)$ are specified, the Gaussian process can be evaluated at a finite number s of input points $\{\mathbf{x}_i\}_{i=1}^s$. The joint distribution of $\{y_i = f(\mathbf{x}_i)\}_{i=1}^s$ follows a s-variate Gaussian distribution. A sample of the s-variate Gaussian distribution, denoted as $\tilde{\mathbf{y}} := (\tilde{y}_1, \ldots, \tilde{y}_s)^T$, can be viewed as a sample of the Gaussian process. It is also

called a realization of the Gaussian process. To generate $\tilde{\mathbf{y}}$, the $s \times s$ covariance matrix $\mathbf{\Sigma}$ needs to be decomposed into the product of a lower triangular matrix $\mathbf{L}$ and its conjugate transpose $\mathbf{L}^T$ using the Cholesky decomposition

$$\mathbf{\Sigma} = \begin{pmatrix} C(\mathbf{x}_1, \mathbf{x}_1) & C(\mathbf{x}_1, \mathbf{x}_2) & \dots & C(\mathbf{x}_1, \mathbf{x}_s) \\ C(\mathbf{x}_2, \mathbf{x}_1) & C(\mathbf{x}_2, \mathbf{x}_2) & \dots & C(\mathbf{x}_2, \mathbf{x}_s) \\ \vdots & \vdots & \ddots & \vdots \\ C(\mathbf{x}_s, \mathbf{x}_1) & \dots & \dots & C(\mathbf{x}_s, \mathbf{x}_s) \end{pmatrix} = \mathbf{L}\mathbf{L}^T. \tag{3.23}$$

Then a sample $\tilde{\mathbf{y}}$ can be obtained as follows:

$$\tilde{\mathbf{y}} = (m(\mathbf{x}_i), \dots, m(\mathbf{x}_s))^T + \mathbf{L}\mathbf{w}, \tag{3.24}$$

where $\mathbf{w} := (w_1, \dots, w_s)^T$ denotes a s-dimensional random vector consisting of s independent standard normal random variables $w_i, i \in \{1, \dots, s\}$.

Similarly, a sample $\tilde{\mathbf{y}}$ from a Student's-t process with a mean function $m(\cdot)$, covariance function $C(\cdot, \cdot)$, and degree of freedom v, can be drawn as follows:

$$\tilde{\mathbf{y}} = (m(\mathbf{x}_i), \dots, m(\mathbf{x}_s))^T + \sqrt{v/\chi^2_v}\mathbf{L}\mathbf{w}, \tag{3.25}$$

where χ^2_v represents an independent random variable following a chi-squared distribution with v degrees of freedom. Details on above-mentioned sampling techniques can be found in Hofert (2013).

In the context of GP emulation, multiple samples (or realizations) are usually drawn from the GP posterior (either a Gaussian process or Student's-t process) by matrix decomposition-based techniques. The samples are then used in any following analysis to take the emulation uncertainty into account. In chapter 4, equation 3.25 will be used to generate multiple samples from GP emulators in order to account for emulation uncertainty in emulator-based global sensitivity analyses. Regarding generating realizations in geostatistical applications, the above matrix decomposition-based sampling techniques may become problematic since s can be very large there and the computational complexity of the Cholesky decomposition is $\mathcal{O}(s^3)$. Many methods that reduce the computational complexity have been proposed and widely used in geostatistical applications, see Liu et al. (2019) for a comprehensive review. In chapter 6, one of these methods called sequential Gaussian simulation will be discussed in detail in the context of topographic uncertainty quantification.

3.5 A simple example

In order to give a direct impression on how GP regression works, a simple example is shown in figure 3.1. Let's assume that the one-dimensional function $y = x + 3\sin(x/2)$ is expensive to query and we only know its values at five locations. The task of GP regression is to approximate the (assumed unknown) function based on the training data. As presented in section 3.2.1, we first need to determine the structure of the mean function and covariance function. For this simple example,

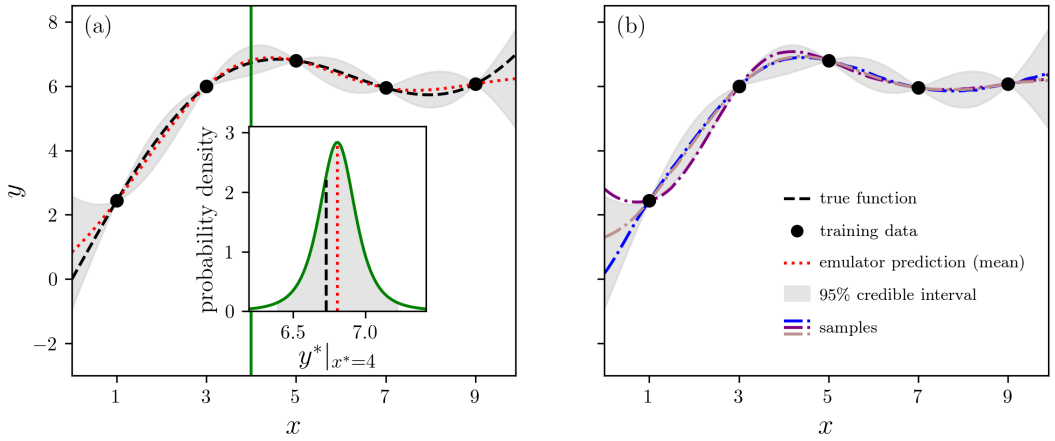

FIGURE 3.1: An example of using GP regression to approximate a simple one-dimensional function $y = x + 3\sin(x/2)$ (O'Hagan, 2006). The black dots represent the training data. The hyperparameters are learned using the maximum marginal likelihood estimation as introduced in section 3.3.3. The resulted GP posterior is a Student's-t process. The inserted plot in panel (a) shows the Student's-t distribution at an untried input point $x^* = 4$ (location noted by the vertical green line); the mean is given by equation 3.21b and the 95% credible interval can be computed based on equations 3.21a-3.21d. Panel (b) shows three samples drawn from the Student's-t process, based on equation 3.25.

the mean is simply modeled with a constant basis function. It means that the regression parameter vector $\boldsymbol{\beta}$ only has one element. The covariance is assumed to be stationary and modeled by equations 3.3-3.4. The next step is to learn the hyperparameters using the training data. All the techniques introduced in section 3.3 can be applied. For the illustration purpose, the technique presented in section 3.3.3 is used here. Namely, the hyperparameters $\boldsymbol{\beta}$ and σ^2 are integrated out and $\boldsymbol{\psi}$ is estimated by maximizing the marginal likelihood. The resulted GP posterior is a Student's-t process, as shown in figure 3.1a. The emulator prediction is given by the mean, namely equation 3.21b. The uncertainty of the emulator prediction can be assessed by for example the 95% credible interval, which can be computed based on equations 3.21a-3.21d. Figure 3.1b shows three samples from the GP posterior. They are drawn from the Student's-t process using equation 3.25.

3.6 Extension to multiple outputs

Till now, I have focused on GP regression for a function $y = f(\mathbf{x})$ with a p-dimensional input $\mathbf{x}$ and a scalar output y. However, there are also cases where a concerned function may have multiple outputs. For example, one may be interested in multiple quantities at spatial locations in geostatistical applications. Also, many simulators produce multiple outputs. Techniques for multi-output regression are often known as cokriging in geostatistics (Álvarez and Lawrence, 2011) and multivariate emulation in the context of surrogate modeling (Rougier et al., 2009). As multi-output cases will only be encountered in the context of surrogate modeling in this thesis,

techniques for multivariate GP emulation will be discussed. For cokriging, one can refer to Cressie (1993).

Let $\mathbf{f}(\mathbf{x})$ denote a simulator with a p-dimensional input $\mathbf{x} = (x_1, \ldots, x_p)^T \in \mathcal{X} \subset \mathbb{R}^p$ and a k-dimensional output $\mathbf{y} = (y_1, \ldots, y_k)^T \in \mathbb{R}^k$. For instance, $\mathbf{f}(\mathbf{x})$ could be a landslide run-out model with $\mathbf{x}$ consisting of its uncertain inputs (like the release volume, rheological parameters, etc) and $\mathbf{y}$ being maximum flow velocity over time at k locations. The most straightforward approach to emulate the multi-output simulator $\mathbf{f}(\mathbf{x})$ is to build an independent scalar GP emulator for each component of the simulator $y_j = f_j(\mathbf{x}), j \in \{1, \ldots, k\}$. This type of approach is sometimes called Many Single emulator approach (Gu and Berger, 2016). Each independent emulator $\hat{f}_j(\mathbf{x})$ has its own hyperparameters $\boldsymbol{\beta}_j$, σ_j^2, and $\boldsymbol{\psi}_j$ which need to be learned using techniques introduced in section 3.3. As a consequence, this type of approach may take a lot of time for training the emulators when k is large.

Alternative multivariate GP emulators can be found in the literature, for example the principal component emulator (Higdon et al., 2008), the outer-product emulator (Rougier, 2008), the parallel partial GP emulator (Gu and Berger, 2016), etc. In this thesis, the parallel partial GP emulator will be used for multi-output cases (see chapter 4). This approach allows directly emulating the mapping from the p-dimensional input to the k-dimensional output. It assumes that each component of the simulator, namely $y_j = f_j(\mathbf{x}), j \in \{1, \ldots, k\}$, follows an independent Gaussian process like in the Many Single emulator approach. The major difference is that it further assumes all k emulators have different regression parameters $\boldsymbol{\beta}_j$ and variances $\sigma_j^2, j \in \{1, \ldots, k\}$, but share the same correlation length parameters $\boldsymbol{\psi}$ and basis functions $\mathbf{h}(\mathbf{x})$. Since the regression parameters and variance can be analytically integrated out (see section 3.3.3), the emulator training reduces to estimate the common correlation length parameters $\boldsymbol{\psi}$, which can be estimated from the overall likelihood (Gu and Berger, 2016). Plugging the estimation of the common $\boldsymbol{\psi}$ into each independent emulator leads to the trained parallel partial GP emulator. It consists of k Student's-t processes, denoted as $\{\hat{f}_j(\mathbf{x})\}_{j=1}^k$. This approach can greatly reduce the time for emulator training.

Chapter 4

Emulator-based global sensitivity analysis of landslide run-out models

4.1 Literature review and motivation

As introduced in sections 2.3-2.4, landslide run-out models require a variety of input data, including initial distribution of the flow mass (a release polygon given as a shape file or a raster map of release heights), topographic data (a digital elevation model), and rheological parameters (dry-Coulomb friction and turbulent friction parameters for the Voellmy rheology). These input data often involve many uncertainties. It is therefore essential to study the model's sensitivity to the uncertain inputs in order to improve our understanding of the computational landslide run-out models and provide guidelines for their future usage.

Sensitivity analyses on landslide run-out models are commonly based upon local one-at-a-time approaches. This type of approach changes one input variable at a time while keeping others at their baseline values to explore its isolated effect on model outputs. For example, Borstad and McClung (2009) and Moretti et al. (2015) studied the sensitivity of the run-out model employing the Coulomb-type rheological model to the Coulomb friction coefficient and initial condition of the release mass, based on a hypothetical parabolic slope and a real rockslide-debris flow event respectively. Both found model outputs are more sensitive to the Coulomb friction coefficient than initial condition of release mass. In terms of the Voellmy rheological model, Barbolini et al. (2000) and Schraml et al. (2015) studied the sensitivity of model outputs to the two Voellmy friction coefficients and initial condition of release mass, while Hussin et al. (2012) studied the sensitivity of model outputs to the two Voellmy friction coefficients and the entrainment coefficient. A common finding is that the run-out distance is mainly influenced by the Coulomb friction coefficient; Barbolini et al. (2000) reported that the release area generally has a lower influence than the other parameters and Schraml et al. (2015) found the release volume causes little variation of the output of RAMMS-DF; Hussin et al. (2012) found the turbulent

friction coefficient has the strongest impact on the maximum flow velocity at control points. Similar one-at-a-time sensitivity analyses of run-out models employing other friction laws, such as the Pouliquen or Mohr-Coulomb law, can be found in Pirulli and Mangeney (2008) and Fathani et al. (2017).

While straightforward to implement, this type of local sensitivity analysis method cannot assess potential interactions between input variables. Their results may highly depend on chosen baseline values (Girard et al., 2016). In contrast, variance-based global sensitivity analyses can fully explore the input space, quantify the contribution of each variable to the output variation, and identify interactions between different variables. The Sobol' method, one typical variance-based method, has been developed and widely used since 1990s (Sobol', 1993; Sobol', 2001; Saltelli, 2002; Saltelli et al., 2010). The key idea of a Sobol' sensitivity analysis is that the variance of a model output can be quantitatively decomposed into contributions due to the independent effect of every single input factor and combined effects of input factors. These are represented by first-order and higher-order Sobol' indices respectively. The Sobol' indices can therefore be interpreted as measures of relative sensitivity. They allow us to identify coupled effects between the various model inputs. The calculation of Sobol' sensitivity indices usually requires Monte Carlo-based methods, leading to a large number of necessary model evaluations. For computationally demanding models, the calculation may be prohibitively expensive. In that case, it is rather promising to employ emulation techniques to overcome the computational challenge.

Gaussian process emulation, as comprehensively introduced in chapter 3, has been utilized for the purpose of global sensitivity analyses in different fields (Lee et al., 2011; Rohmer and Foerster, 2011; Lee et al., 2012; Bounceur et al., 2015; Girard et al., 2016; Aleksankina et al., 2019). These studies either focus on emulating the evaluation of a few scalar outputs (Lee et al., 2011; Rohmer and Foerster, 2011; Girard et al., 2016), or build separate emulators for each of the many outputs using the Many Single emulator approach (Lee et al., 2012; Aleksankina et al., 2019). One exception among them is Bounceur et al. (2015), who combined emulation techniques with the principal component analysis leading to emulation of a reduced-order model (namely the principal component emulator). For a simulator with massive outputs like a landslide run-out model, building separate emulators for each output can be computationally intensive (Gu and Berger, 2016). In recent years, great improvement has been made to enable simultaneous emulation for multi-output models, see section 3.6.

The goal of this chapter is twofold. The first is a methodological goal, namely to develop a new methodology that enables efficient global sensitivity analyses of landslide run-out models, by integrating recent development of landslide run-out models (Mergili et al., 2017), GP emulation techniques (Gu and Berger, 2016; Gu et al., 2018; Gu et al., 2019), and global sensitivity analyses (Le Gratiet et al., 2014). The second goal is application-oriented and aims at employing the methodology to

assess the relative importance of different uncertain inputs, specifically rheological parameters and the release volume, and their interactions in landslide run-out models based on the 2017 Bondo landslide event.

4.2 Methodology

4.2.1 Sobol' sensitivity analysis

Let $f(\mathbf{x})$ denote a simulator with a p-dimensional input $\mathbf{x} = (x_1, \ldots, x_p)^T \in \mathcal{X} \subset \mathbb{R}^p$ and a scalar output $y \in \mathbb{R}$. Input uncertainties of $\mathbf{x}$ induce output uncertainty of y. The essential idea of a Sobol' sensitivity analysis is to decompose the variance of y into contributions caused by each x_i and their interactions. In practice, p first-order indices $\{S_i\}_{i=1}^p$ and p total-effect indices $\{S_{Ti}\}_{i=1}^p$ are usually computed. They are defined as (Saltelli et al., 2010)

$$S_i = \frac{V_{x_i}(E_{\mathbf{x}_{-i}}(y|x_i))}{V(y)}, \tag{4.1a}$$

$$S_{Ti} = 1 - \frac{V_{\mathbf{x}_{-i}}(E_{x_i}(y|\mathbf{x}_{-i}))}{V(y)}, \tag{4.1b}$$

where V and E represent the variance and expectation operator respectively, $\mathbf{x}_{-i}$ denotes the vector consisting of all input factors except x_i. A first-order index S_i accounts for the contribution of the input factor x_i to the variance of the output, independent from other input factors $\mathbf{x}_{-i}$; a total-effect index S_{Ti} indicates the total contribution of x_i to the output variation, i.e. the sum of its first-order contribution and all high-order effects owing to interactions (Saltelli et al., 2008). The difference $S_{Ti} - S_i$ thus indicates any interaction between x_i and $\mathbf{x}_{-i}$. Employing this concept to landslide run-out models will hence allow us to investigate combined effects of for example the rheological parameters and the release volume on simulation outputs.

Computing the conditional variances in equations 4.1a-4.1b involves nested integrals (Girard et al., 2016). This is analytically impractical for complex simulators like landslide run-out models. Instead, Monte Carlo-based methods are commonly used to estimate the Sobol' indices. The uncertainty introduced by Monte Carlo-based integration can be taken into account using a bootstrap strategy (Archer et al., 1997).

In this chapter, the numerical procedure presented in Saltelli et al. (2010) is employed. The computational cost is $N \cdot (p + 2)$ evaluations of a simulator, where N is the base sample size. More specifically, the denominator $V(y)$ in equations 4.1a-4.1b can be estimated using $2 \cdot N$ simulation runs based on two independent sets of input points. Each set consists of N input points for the simulator. Moreover, each pair of numerators in equations 4.1a-4.1b requires additional N simulation runs corresponding to a new set of N input points, which is constructed from the two independent sets. It leads to additional $p \cdot N$ simulation runs. For the detailed procedure, please refer to Saltelli et al. (2010).

Algorithm 1 Emulator-based Sobol' index evaluation

1: Choose input points $\mathbf{x}^{tr} = \{\mathbf{x}_i\}_{i=1}^{n^{tr}}$, usually using a Latin hypercube design.
2: Run the simulator $f(\mathbf{x})$, e.g. the landslide run-out model, at each of n^{tr} chosen input points to obtain outputs $\mathbf{y}^{tr}$.
3: Build the emulator $\hat{f}(\mathbf{x})$ based on $\{\mathbf{x}^{tr}, \mathbf{y}^{tr}\}$ (sections 3.2-3.3).
4: Set the base sample size N, realization sample size N_r, and bootstrap sample size N_b. Sample inputs $\{\mathbf{x}_i\}_{i=1}^{N\cdot(p+2)}$ from the input domain according to Saltelli et al. (2010) (section 4.2.1).
5: **for** $n_r = 1, \ldots, N_r$ **do**
6: Sample a realization at input points $\{\mathbf{x}_i\}_{i=1}^{N\cdot(p+2)}$ from the emulator $\hat{f}(\mathbf{x})$, denoted as $\{\tilde{f}^{n_r}(\mathbf{x}_i)\}_{i=1}^{N\cdot(p+2)}$ (section 3.4).
7: Compute $\{\hat{S}_i^{n_r,1}\}_{i=1}^{p}$ and $\{\hat{S}_{Ti}^{n_r,1}\}_{i=1}^{p}$ based on $\{\tilde{f}^{n_r}(\mathbf{x}_i)\}_{i=1}^{N\cdot(p+2)}$.
8: **for** $n_b = 2, \ldots, N_b$ **do**
9: Sample with replacements $\{\tilde{\mathbf{x}}_i\}_{i=1}^{N\cdot(p+2)}$ from $\{\mathbf{x}_i\}_{i=1}^{N\cdot(p+2)}$ and record the new realization $\{\tilde{f}^{n_r}(\tilde{\mathbf{x}}_i)\}_{i=1}^{N\cdot(p+2)}$.
10: Compute $\{\hat{S}_i^{n_r,n_b}\}_{i=1}^{p}$ and $\{\hat{S}_{Ti}^{n_r,n_b}\}_{i=1}^{p}$ based on the new realization $\{\hat{f}^{n_r}(\tilde{\mathbf{x}}_i)\}_{i=1}^{N\cdot(p+2)}$.
11: **end for**
12: **end for**
13: **return** $\{\hat{S}_i^{n_r,n_b}\}_{i=1,\ldots,p}^{n_r=1,\ldots,N_r;n_b=1,\ldots,N_b}$ and $\{\hat{S}_{Ti}^{n_r,n_b}\}_{i=1,\ldots,p}^{n_r=1,\ldots,N_r;n_b=1,\ldots,N_b}$.
14: Estimate S_i and S_{Ti} defined in equations 4.1a-4.1b using $\hat{S}_i = \frac{1}{N_r \cdot N_b}\sum \hat{S}_i^{n_r,n_b}$ and $\hat{S}_{Ti} = \frac{1}{N_r \cdot N_b}\sum \hat{S}_{Ti}^{n_r,n_b}$, with $i = 1, \ldots, p$. Quantify the overall uncertainty (i.e. Monte Carlo-based sampling uncertainty and emulator uncertainty) of an estimated Sobol' index using its standard deviation or 95% credible interval.

As pointed out in Saltelli et al. (2010), N should be sufficiently large, e.g. 500 or higher, which is critical in our case as the landslide run-out model itself is computationally intensive. If a single run of the landslide run-out model described in section 2.2 costs 32 minutes, which correspond to the average run time of the 200 simulation runs in section 4.4.3, the sensitivity analysis for three input variables will cost at least $32 \times 500 \times (3+2) = 80000$ minutes, roughly 56 days on a single core. Therefore, it is necessary to employ emulation techniques to improve the computational efficiency in order to carry out this type of global sensitivity analysis.

4.2.2 Emulator-based Sobol' sensitivity analysis

The idea of emulator-based Sobol' sensitivity analysis is to train GP emulators for the expensive-to-run landslide run-out model using a small number of model evaluations. Then the cheap-to-run GP emulators are used to conduct the Sobol' sensitivity analysis instead of the original simulator. However, the efficiency improvement by using GP emulation comes at a cost, i.e. additional emulator uncertainty. This type of uncertainty can be quantified as it can be evaluated from the emulator directly, see section 3.2.2. Yet, methods to account for this uncertainty in the subsequent analysis

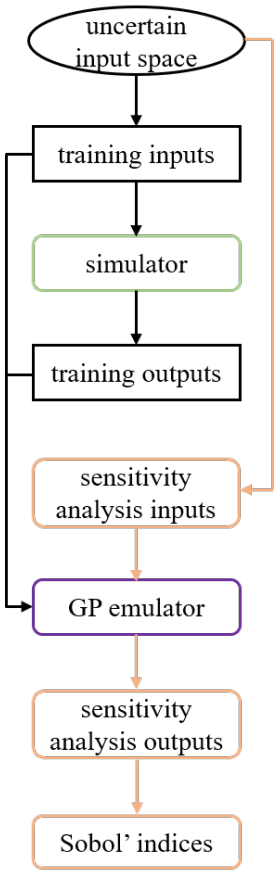

FIGURE 4.1: A schematic representation of the emulator-based global sensitivity analysis method.

are needed. Alongside the development of emulation techniques and global sensitivity analysis methods, a number of methods have been developed in recent years to address this issue in global sensitivity analyses, e.g. Oakley and O'Hagan (2004), Marrel et al. (2009), Janon et al. (2014), and Le Gratiet et al. (2014).

The method proposed by Le Gratiet et al. (2014) is used in this thesis. It combines the work of Oakley and O'Hagan (2004) and Janon et al. (2014). This method can simultaneously take the Monte Carlo-based sampling uncertainty (section 4.2.1) and emulator uncertainty into account when calculating the Sobol' indices. In this thesis, it is adapted to combine the sampling scheme presented in Saltelli et al. (2010) and the GP emulators developed by Gu and Berger (2016) and Gu et al. (2018).

The adapted method for a simulator with a scalar output, namely $f(\mathbf{x})$, is shown in Algorithm 1. A schematic representation is illustrated in figure 4.1. For a simulator with a k-dimensional output, i.e. $\mathbf{f}(\mathbf{x})$, the method is essentially similar. Minor modifications are as follows.

- In steps 1-3, a parallel partial GP emulator $\{\hat{f}_j(\mathbf{x})\}_{j=1}^{k}$ is built (section 3.6) instead of a scalar GP emulator $\hat{f}(\mathbf{x})$.
- Steps 5-14 are repeated for each $\hat{f}_j(\mathbf{x})$ to evaluate the Sobol' indices at the j-th element of the k-dimensional output, where $j = 1, \ldots, k$.

4.3 Implementation

The methodology proposed in section 4.2.2 consists of several components, including the Voellmy-type landslide run-out model (section 2.2), multivariate GP emulation (section 3.6), Sobol' sensitivity analysis (section 4.2.1), and an algorithm that accounts for emulator uncertainty in Sobol' sensitivity analysis. The implementation relies on open-source software and packages that have been recently developed for each component. It should be noted that although these individual building blocks exist to date, they do not interact seamlessly as of now. A software framework that allows us to efficiently couple and leverage these building blocks together does not exist. Our Python-based implementation provides such a framework. Its benefit is that only one controlling Python script is required to automatically run simulations at design points based on a Latin hypercube design (see section 4.4.3), construct GP emulators, and conduct Sobol' sensitivity analysis. It coordinates the individual building blocks which involve different programming languages and dependencies, from within a single Python environment. It therefore automatizes the workflow, reduces the redundant manual and potentially error-prone data format transformation between different software and packages, and minimizes the requirement of users' knowledge on the dependent software and packages. The principal components of the implementation are as follows:

- **Simulator**. Mergili et al. (2017) presented the open-source software r.avaflow for simulation of a variety of mass flows, which relies on GRASS GIS 7. It employs a Voellmy-type model (section 2.2) and a multi-phase mass flow model (Pudasaini and Mergili, 2019). Here, the former is the simulator under investigation. I implemented a Python-based wrapper to automatically prepare a batch job, run simulations, and extract outputs given the selected values of input variables $\mathbf{x}^{tr}$, without explicitly starting GRASS and r.avaflow.

- **Emulator**. Gu et al. (2019) presented the R package RobustGaSP (Robust Gaussian Stochastic Process Emulation), in which they implemented the maximum likelihood estimation (section 3.3.1), maximum marginal likelihood estimation (integrating out $\boldsymbol{\beta}$ and σ^2; section 3.3.3), and the maximum marginal posterior estimation (section 3.3.3) for learning the hyperparameters. They also implemented the parallel partial GP emulator (section 3.6). I implemented a Python-based wrapper based on rpy2 (the Python interface to the R language) to utilize RobustGaSP within the unified Python-based framework.

- **Emulator-based Sobol' analysis**. Herman and Usher (2017) presented the Python package SALib (Sensitivity Analysis Library in Python), in which the numerical procedure of calculating the Sobol' indices for a simulator is implemented. I extended their codes to realize Algorithm 1 which enables emulator-based Sobol' analysis for multi-output simulators.

It should be noted that our Python-based framework is implemented in a modular way. The sensitivity of any other landslide run-out model can therefore be studied using our workflow by simply replacing the simulator.

4.4 Case study based on the 2017 Bondo landslide event

4.4.1 Case background

Pizzo Cengalo (see figure 4.2) located in the Swiss Alps, is subjected to rock fall and landslide events since decades due to its geological pre-conditioning factors (Walter et al., 2020). Two recent landslide events in that area are well-documented and widely studied. The first event occurred on December 27th 2011. Around 1.5 million m^3 of rock detached from the northeastern face of Pizzo Cengalo and evolved into a rock avalanche traveling 2.7 km down the Bondasca valley. The second event occurred on August 23th 2017. Approximately 3 million m^3 of rock were released from the northeastern face of Pizzo Cengalo, leading to a rock avalanche traveling 3.2 km down the Bondasca valley. A part of the rock avalanche turned into an initial debris flow, followed by a series of additional debris flows within 48 hours, which reached the village Bondo (Walter et al., 2020).

The case study in this chapter is based on the topography and release area of the 2017 landslide event. A pre-event digital elevation model (DEM) and a post-event DEM are available, both with 1 m resolution. They are based on airborne laser scans after the 2011 and after the 2017 events, as well as aerial images acquired by the Swiss topographic services Swisstopo (Walter et al., 2020). Release area and initial mass distribution of the event can be obtained from the height difference map of the two DEMs. As the topographic input, we use a merged DEM based on the pre-event and post-event DEMs. The merged DEM reflects the post-event topography in the release area and pre-event topography in other areas. In addition, we use the same release area as the 2017 landslide event, as shown in figure 4.2. The grid size of the computational mesh for the simulator is set to be 10 m.

It should be noted that the intention of the case study is not to back-analyze the 2017 landslide event. Other publications are devoted to that research question (Mergili et al., 2020; Walter et al., 2020). The focus here is to apply the novel emulator-based global sensitivity analysis to the Bondo event in order to assess the model's sensitivity to Voellmy rheological parameters μ and ξ, and the release volume v_0. Here, the quantities of interest obtained by post-processing landslide run-out model outputs are:

- Angle of reach, the tangent of which equals to the ratio of the landslide fall height and projected run-out distance, namely the Heim's ratio (Lucas et al., 2014). The angle of reach generally decreases as the run-out distance increases.
- Impact area, defined as the area of the region where maximum flow height values exceed a threshold value, here 0.1 m.

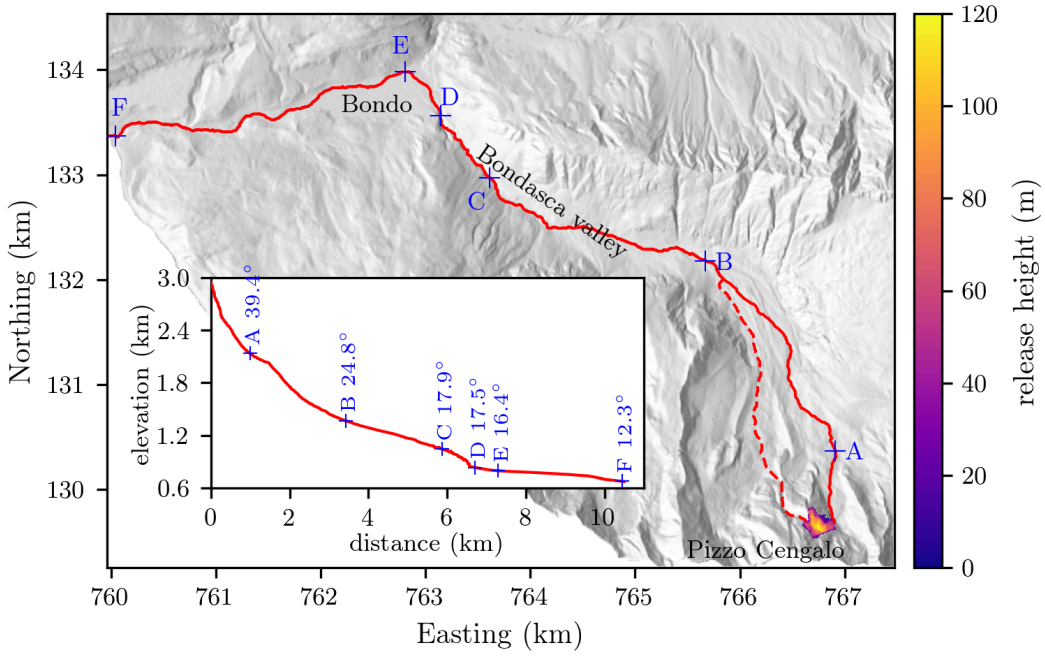

FIGURE 4.2: Pizzo Cengalo-Bondo topography. The colormap shows the distribution of the release mass of the 2017 landslide event. The solid line and dashed line denote the major and minor flow paths. The embedded plot in the bottom-left corner shows the profile of the major flow path, on top of which locations A-F with respective angle of reach are noted for later discussions in section 4.5.2.

- Maximum flow height at k locations, denoted as $(h_{l_1}^{\max}, \ldots, h_{l_k}^{\max})^T$.
- Maximum flow velocity at k locations, denoted as $(\|\mathbf{u}_{l_1}\|^{\max}, \ldots, \|\mathbf{u}_{l_k}\|^{\max})^T$.

4.4.2 Ranges of uncertain inputs

Sosio et al. (2008) summarized typical ranges for μ and ξ based on a variety of literature. For rock avalanches and debris flows, the range for μ is 0.05-0.25 and that for ξ is 200-1000 $\mathrm{m/s^2}$. Schraml et al. (2015) presented many back-analyzed μ-ξ sets, consisting of published values in the literature and their own case study. For most of the rock avalanche and debris flow events, μ lies within the range 0.02-0.25 and ξ varies between 100-2000 $\mathrm{m/s^2}$. Aaron and McDougall (2019) presented back-analyses results of a rock avalanche dataset consisting of 45 past rock avalanche events. Their calibrated values of μ vary between 0.025-0.29, except 4 cases in which the path material is bedrock. The calibrated values of ξ are in the range 200-2100 $\mathrm{m/s^2}$.

Based on the reference studies, the ranges 0.02-0.3 and 100-2200 $\mathrm{m/s^2}$ are set for μ and ξ respectively. The release volume v_0 is assumed to vary between 1.5 million $\mathrm{m^3}$ and 4.5 million $\mathrm{m^3}$, namely $\pm 50\%$ based on the 3 million $\mathrm{m^3}$ release volume of the 2017 landslide event. This is achieved by multiplying the distribution of initial mass of the 2017 landslide event with a value between 0.5 and 1.5. To sum up, the three uncertain inputs result in a three dimensional input space, where μ, ξ, and v_0 vary independently within 0.02-0.3, 100-2200 $\mathrm{m/s^2}$, and 1.5-4.5 million $\mathrm{m^3}$.

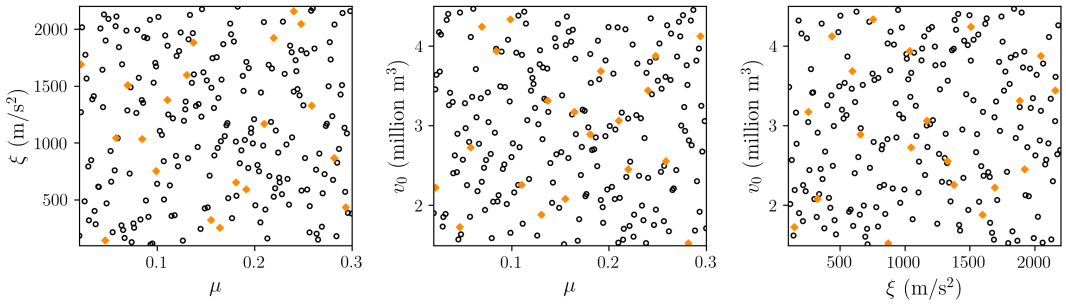

FIGURE 4.3: Two-dimensional projection of the 200 training samples (void circles) and 20 validation samples (solid diamonds) from two independent maximin Latin hypercube designs. Left ξ-μ, middle v_0-μ, right v_0-ξ. The 200 samples are used to build the emulators. The 20 samples are used to validate the parallel partial GP emulators.

4.4.3 Emulator design and validation

To prepare the emulator training data, $n^{tr} = 200$ samples are drawn from the three dimensional input space using the maximin Latin hypercube design which maximizes the minimum distance between design points to achieve optimum space-filling properties (Aleksankina et al., 2019), see figure 4.3. This results in $\mathbf{x}^{tr} = \{(\mu_i, \xi_i, v_{0i})^T\}_{i=1}^{200}$. One run-out simulation takes 32 minutes on average on a laptop with Intel Core i7-9750H CPU. For each simulation run, the angle of reach, impact area, $(h_{l_1}^{\max},\ldots,h_{l_k}^{\max})^T$ and $(\|\mathbf{u}_{l_1}\|^{\max},\ldots,\|\mathbf{u}_{l_k}\|^{\max})^T$ at $k = 47958$ chosen locations are extracted. At each of the 47958 locations, at least one of the 200 simulation runs has a maximum flow height value larger than 0.1 m. Correspondingly, two scalar GP emulators (section 3.3.3) and two parallel partial GP emulators (section 3.6) are built based on $\mathbf{x}^{tr}$ and its respective simulation outputs. Each parallel partial GP emulator takes about 0.05 seconds to predict maximum flow height or velocity at all 47958 locations for a new input point.

Before using the emulators for further sensitivity analyses, their performance needs to be validated. The proportion of validation outputs that lie in emulator-based 95% credible intervals is chosen as the diagnostic, denoted as $P_{\text{CI(95\%)}}$. This is commonly used in the literature (e.g. Lee et al., 2011; Spiller et al., 2014; Bounceur et al., 2015; Gu and Berger, 2016). It is defined as

$$P_{\text{CI(95\%)}} = \frac{1}{n}\sum_{i=1}^{n} 1\{f(\mathbf{x}_i^*) \in \hat{f}(\mathbf{x}_i^*)_{\text{CI(95\%)}}\}, \tag{4.2}$$

where n is the number of input points for validation, $f(\mathbf{x}_i^*)$ and $\hat{f}(\mathbf{x}_i^*)_{\text{CI(95\%)}}$ denote the simulation output and the 95% credible interval of the emulator prediction at the input $\mathbf{x}_i^*$ respectively. $P_{\text{CI(95\%)}}$ would be close to 0.95 for an ideal emulator.

The two scalar emulators are validated using the leave-one-out cross validation method as implemented in the RobustGaSP package (meaning $n = 200$), see figure 4.4. Both emulators perform well with emulator prediction values being close to simulator outputs and $P_{\text{CI(95\%)}}$ close to 0.95. As no cross validation scheme is implemented in the RobustGaSP package for a parallel partial GP emulator, the two

parallel partial GP emulators for $(h_{l_1}^{\max},\ldots,h_{l_k}^{\max})^T$ and $(\|\mathbf{u}_{l_1}\|^{\max},\ldots,\|\mathbf{u}_{l_k}\|^{\max})^T$ are validated using additional 20 simulation runs based on an independent maximin Latin hypercube design, see figure 4.3. Figure 4.5a shows $P_{\text{CI(95\%)}}$ values at each location and their distribution in the form of a box plot based on the maximum flow height emulator. Figure 4.5b shows the same evaluation based on the maximum flow velocity emulator. The lowest $P_{\text{CI(95\%)}}$ value of the maximum flow height/velocity emulator is 0.6/0.65, and 95% of the $P_{\text{CI(95\%)}}$ values of both emulators are within 0.8-1. Both emulators show good performance with mean values of $P_{\text{CI(95\%)}}$ over all locations being 0.93 and 0.94 respectively.

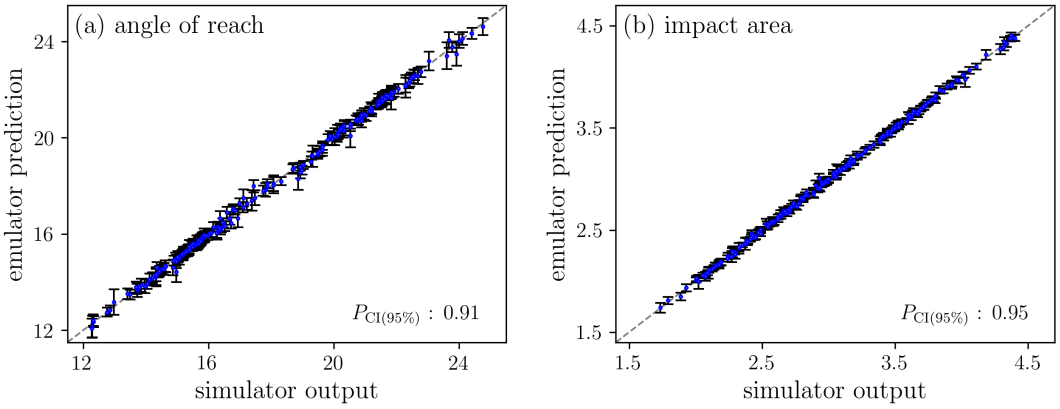

FIGURE 4.4: Leave-one-out cross validation of the GP emulators for scalar outputs (a) angle of reach (in degree), and (b) impact area (in million m^2). The error bars denote 95% credible intervals of the emulator predictions.

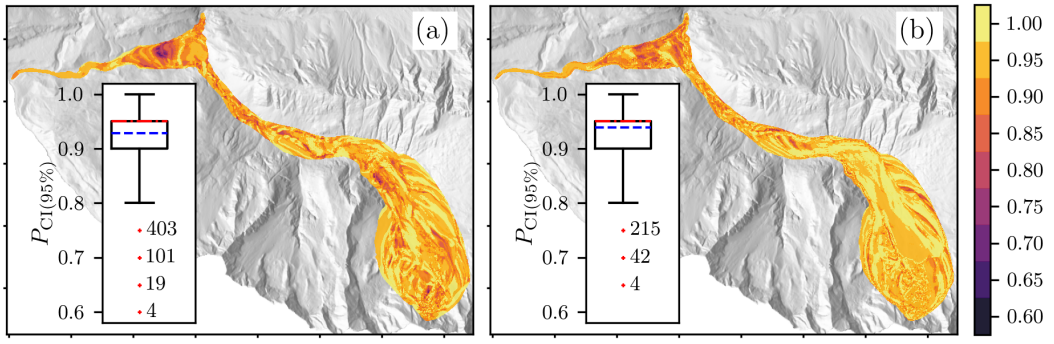

FIGURE 4.5: Validation of the parallel partial GP emulators for vector outputs (a) $(h_{l_1}^{\max},\ldots,h_{l_k}^{\max})^T$ and (b) $(\|\mathbf{u}_{l_1}\|^{\max},\ldots,\|\mathbf{u}_{l_k}\|^{\max})^T$ with $k = 47958$, using 20 validation runs based on an independent maximin Latin hypercube design. In each panel, the colormap shows the $P_{\text{CI(95\%)}}$ values at each location; the box plot presents the distribution of $P_{\text{CI(95\%)}}$ values. In the box plot, the whiskers denote the 2.5th and 97.5th percentiles; the blue dashed line denotes the mean; the number of outliers for each outlier value is noted due to overlapping. The mean of $P_{\text{CI(95\%)}}$ over all locations for maximum flow height/velocity is 0.93/0.94.

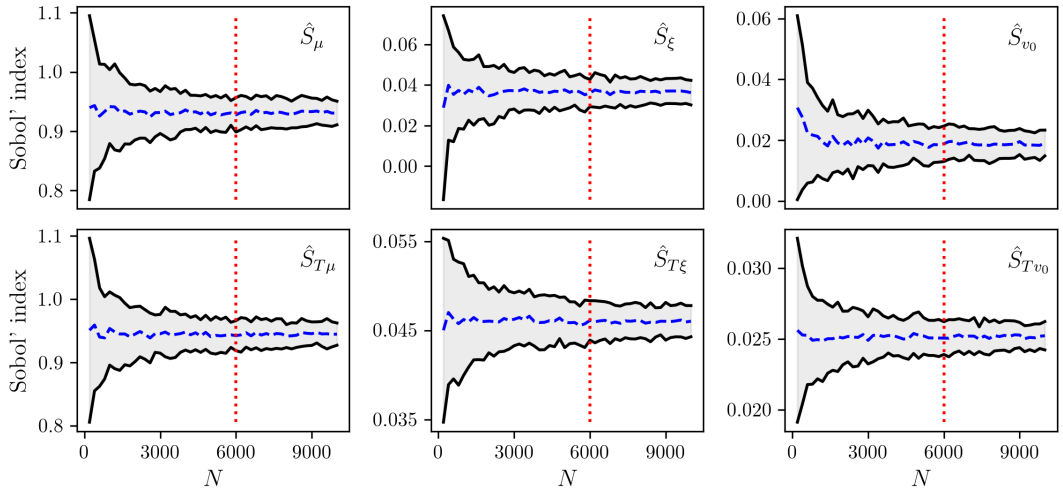

FIGURE 4.6: First-order (first row) and total-effect Sobol' indices (second row) based on the Gaussian process emulator for the angle of reach, with $N_r = N_b = 50$ and N increasing from 200 to 10000 with a step size 200. $\hat{S}_\mu$, $\hat{S}_\xi$, and $\hat{S}_{v_0}$ denote estimated first-order Sobol' indices of μ, ξ, and v_0; $\hat{S}_{T\mu}$, $\hat{S}_{T\xi}$, and $\hat{S}_{Tv_0}$ denote estimated total-effect Sobol' indices of μ, ξ, and v_0 (see step 14 in Algorithm 1 and equations 4.1a-4.1b). In each panel, the dashed line and solid line show the change of the estimated Sobol' index and its 95% credible interval respectively; the estimated Sobol' index tends to converge for $N \geq 4000$ and the length of its 95% credible interval hardly decreases for $N \geq 6000$.

4.4.4 Preliminary convergence analysis

The base sample size N, realization sample size N_r, and bootstrap sample size N_b need to be determined before using the validated emulators for the Sobol' sensitivity analysis (see Algorithm 1). Here, the results of a convergence analysis based on the validated emulator for the angle of reach are presented in order to determine values for these sample sizes. Figure 4.6 shows how the estimated Sobol' indices and their 95% credible interval values change with N increasing from 200 to 10000 with a step size 200, while keeping $N_r = N_b = 50$. It can be seen that the estimated Sobol' indices tend to converge when N is larger than 4000, and their 95% credible interval lengths almost do not decrease for $N \geq 6000$. The same analysis with $N_r = N_b = 100$ and $N_r = N_b = 200$ is also conducted. The results are similar to the findings with $N_r = N_b = 50$, indicating little impact of N_r and N_b. Therefore, $N = 6000$ and $N_r = N_b = 50$ are set for the following sensitivity study. It leads to $N \cdot (p+2) = 6000 \cdot (3+2) = 30000$ samples from the three dimensional input space to estimate the Sobol' indices, namely $\{(\mu_i, \xi_i, v_{0_i})^T\}_{i=1}^{30000}$. Among them, $2 \cdot N = 12000$ samples are used to estimate the overall variance term $V(y)$ in equations 4.1a-4.1b, see section 4.2.1.

4.5 Results and discussions

4.5.1 Angle of reach and impact area

The box plot in figure 4.7a shows the distribution of emulator predicted angle of reach values corresponding to the 12000 samples used to estimate the variance of the

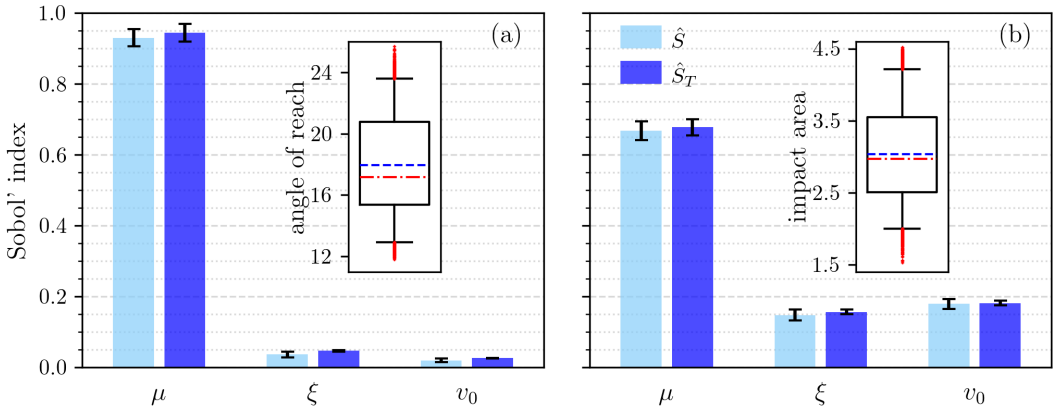

FIGURE 4.7: Sobol' indices for aggregated scalar outputs (a) angle of reach and (b) impact area. The error bars of the bar plots indicate 95% credible intervals of estimated Sobol' indices, which account for Monte Carlo-based sampling uncertainty and emulator uncertainty. The box plots show the distribution of emulator-predicted angle of reach values (in degree) and that of emulator-predicted impact area values (in million m^2). They visualize the variation of the angle of reach and impact area resulting from the uncertain input variables respectively. In each box plot, the whiskers denote the 2.5th and 97.5th percentiles; the blue dashed line denotes the mean; the red dotted dashed line denotes the median; the red crosses denote the outliers.

angle of reach (see section 4.4.4). Due to input uncertainties, the angle of reach could vary in a wide range, around 11.8°-25.7°. The mean is 17.9°. The standard deviation is 3.1° which corresponds to the square root value of $V(y)$ in equations 4.1a-4.1b. The bar plots in figure 4.7a display the estimated first-order and total-effect Sobol' indices, with the 95% credible interval denoting the Monte Carlo-based sampling uncertainty and emulator uncertainty. Each pair of bar plots corresponds to the first-order and total-effect Sobol' indices of one input variable. It is evident that angle of reach is dominated by the dry-Coulomb friction coefficient μ of which the first-order index is over 0.9, whereas both the turbulent friction coefficient ξ and the release volume v_0 show little influence on the angle of reach, with both first-order indices being smaller than 0.05. This result is expected since μ governs the slope angle on which flow mass begins to deposit (McDougall, 2017). It is also consistent with the common finding in former one-at-a-time sensitivity analyses on landslide run-out models employing the Voellmy rheology, such as Barbolini et al. (2000), Hussin et al. (2012), Schraml et al. (2015), and Frey et al. (2016). All of them found that the run-out distance (indicated by the angle of reach) is predominantly affected by the dry-Coulomb friction coefficient μ. In particular, Barbolini et al. (2000) found that there is a difference about half an order of magnitude between the sensitivity of run-out distance to μ and to other parameters like ξ, release height, and release area. Furthermore, it is noteworthy that the difference between the first-order and total-effect indices is small, indicating weak interactions among the three input variables regarding the angle of reach.

Similarly, the box plot in figure 4.7b shows the distribution of emulator predicted

impact area values. Owing to input uncertainties, the impact area could vary between 1.5-4.5 million m^2 with a standard deviation 0.6 million m^2. From the bar plots, it can be seen that estimated first-order indices of μ, ξ, and v_0 are around 0.67, 0.15, 0.18 respectively. It indicates that μ contributes the most to the variance of the impact area, followed by v_0 and ξ. Similar to the results on the angle of reach, the small difference between the first-order and total-effect indices implies that the three input variables barely interact with each other concerning the impact area. Compared to the results of the angle of reach, the importance of μ on the impact area decreases and that of ξ and v_0 increases. A plausible explanation is that the angle of reach only depends on the deposit (assuming that the release area remains the same) where μ plays the dominant role, whereas the impact area depends on all inundated region where all three input variables may have impact.

4.5.2 Maximum flow height and velocity

Before discussing global sensitivity analysis results on maximum flow height and maximum flow velocity, the statistics that are needed to interpret the results are summarized. Figures 4.8a-c show the mean, standard deviation, and coefficient of variation of emulator-predicted maximum flow height values at each location. Figures 4.8d-f show the counterparts of emulator-predicted maximum flow velocity values. The major and minor flow paths as well as locations A-F along the major flow path are noted to facilitate the description of results. The profile of the major flow path and the angle of reach values corresponding to locations A-F are shown in figure 4.2. Location A sits near the release area, where the slope is steep. From location B to location D is the Bondasca valley. Location C corresponds to the mean location of 12000 angle of reach values (17.9°), denoting the average run-out distance. From location D to location E is the debris flow retention basin (Walter et al., 2020). Location F is near the west boundary of the DEM.

It can be seen from figures 4.8a and 4.8d that in general, the mean of maximum flow height gradually decreases along the flow path whereas the mean of maximum flow velocity first increases then decreases reflecting the acceleration and deceleration process. Along the path cross section direction, both the mean of maximum flow height and that of maximum flow velocity generally decrease from the center to the sides. In addition, the mean values in the upstream area of location B are on average much larger than the mean values in the downstream area of location B, possibly because the average slope from the release zone to location B is larger than that beyond location B (see figure 4.2) and the corner around location B decelerates the flow mass.

The standard deviation shown in figures 4.8b and 4.8e reflects the variation of maximum flow height and velocity at each location resulting from uncertainties of the three input variables. It corresponds to the square root of $V(y)$ in equations 4.1a-4.1b. In the Bondasca valley between location B and location D, where the channel is

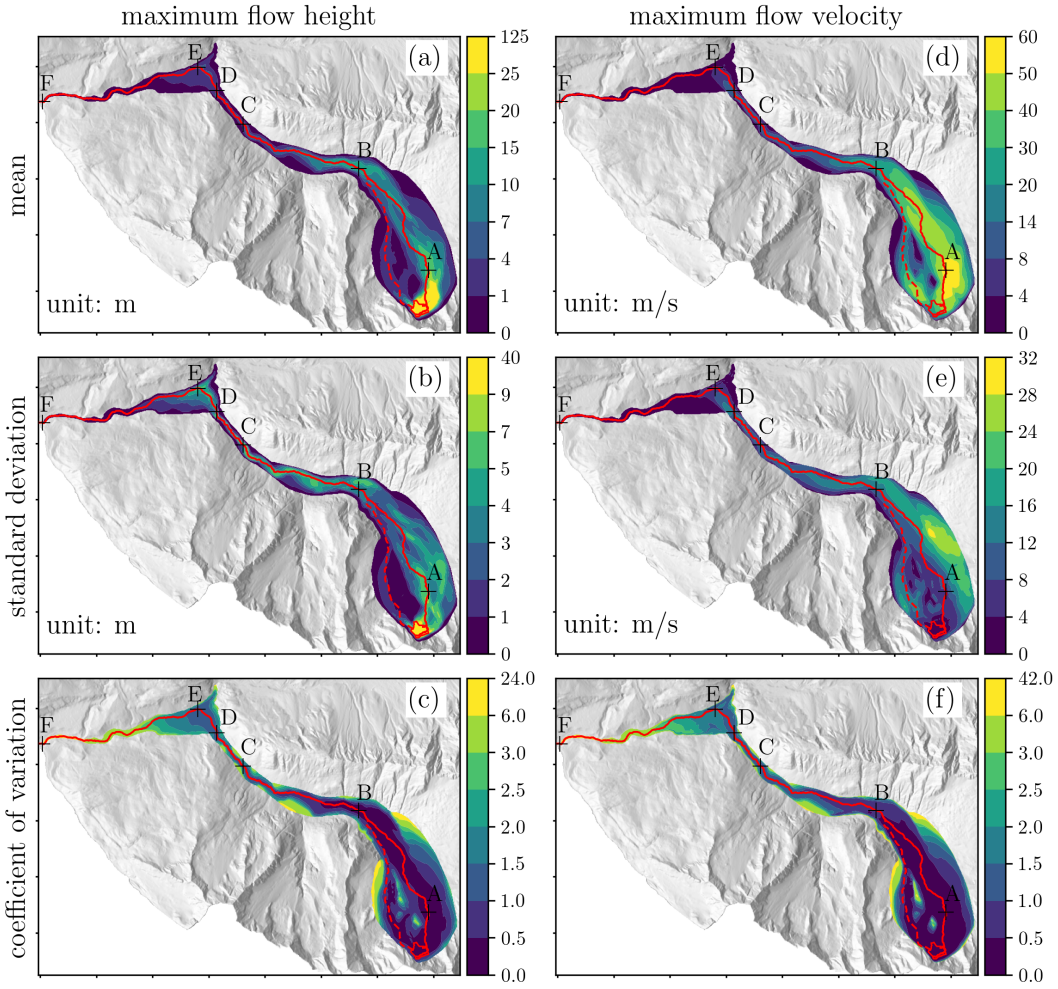

FIGURE 4.8: Statistics of emulator predicted maximum flow height (left column) and velocity (right column) at $k = 47958$ locations. For each location, the mean (first row), standard deviation (second row), and coefficient of variation (third row) are calculated from 12000 emulator predicted maximum flow height and velocity values at that location (see section 4.4.4). The polygon at the bottom-right corner of each panel denotes the release area. The local low/high values on the left side of location A in each panel result from the local ridges (see figure 4.2).

well-defined, the standard deviation generally decreases from the center to the sides in lateral direction, similar to the trend observed in figures 4.8a and 4.8d.

Figures 4.8c and 4.8f present the coefficient of variation defined as the ratio of the standard deviation to the mean, representing the relative variation. Comparing figures 4.8c and 4.8f with figures 4.8a and 4.8d, we find strong negative correlation between the coefficient of variation and the mean. The coefficient of variation generally increases both along the longitudinal direction and from the center to the sides in the lateral direction. A noteworthy feature is that figure 4.8b shows large differences to figure 4.8e, whereas figures 4.8c and 4.8f greatly resemble each other. It indicates that for maximum flow height and velocity, their absolute variation represented by the standard deviation differs from each other, whereas their relative variation represented by the coefficient of variation shows great similarities.

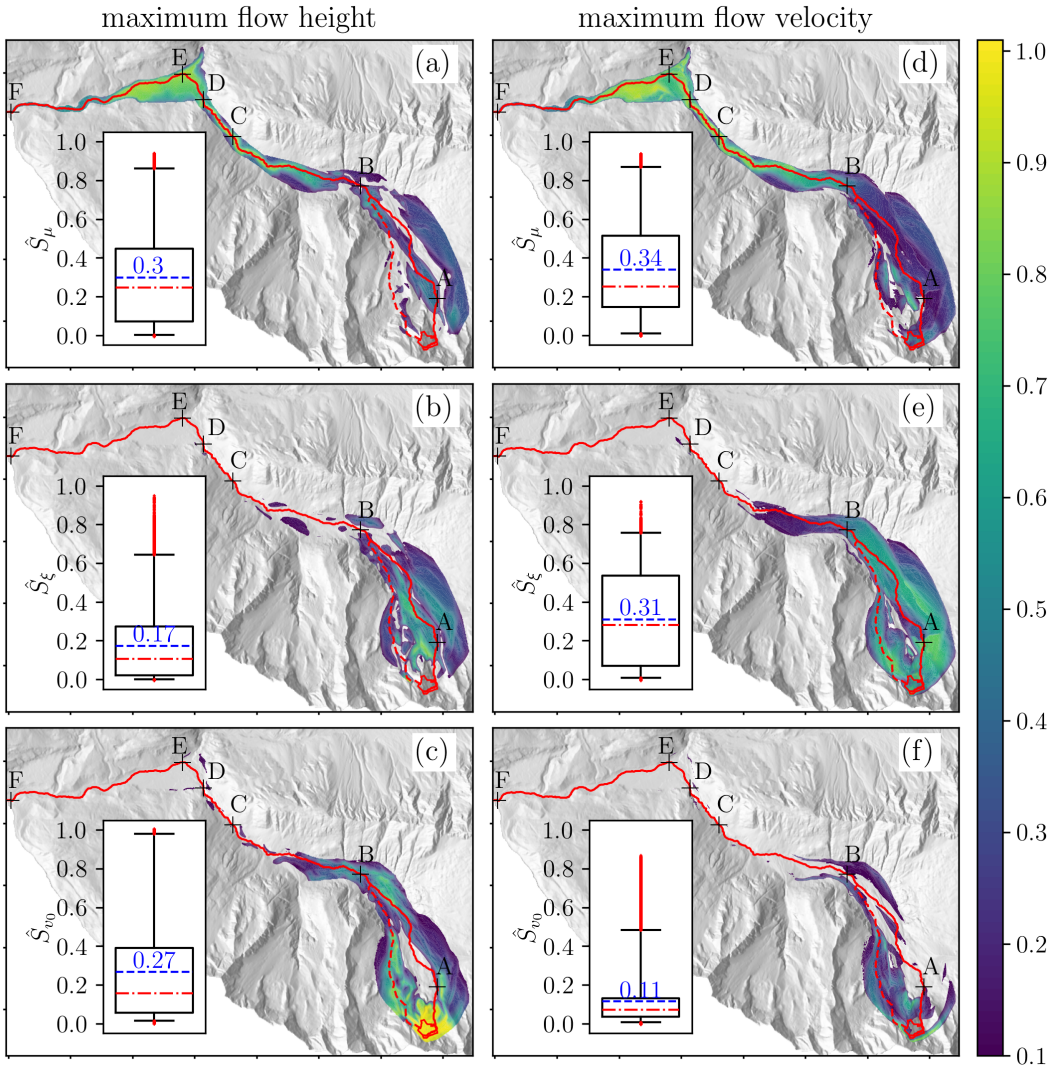

FIGURE 4.9: First-order Sobol' indices for $(h_{l_1}^{\max},\ldots,h_{l_k}^{\max})^T$ (left column) and for $(\|\mathbf{u}_{l_1}\|^{\max},\ldots,\|\mathbf{u}_{l_k}\|^{\max})^T$ (right column). In each panel, values smaller than 0.1 are not shown in the colormap; the box plot presents the distribution of respective first-order indices at all locations (including values smaller than 0.1); the mean over all locations is notated in the box plot.

Figures 4.9-4.10 present results of the Sobol' sensitivity analysis on maximum flow height and velocity at each location. The uncertainties of estimated Sobol' indices are found to be negligible and have little impact on the discussion (see figure 4.7). The 95% credible interval is therefore omitted here to avoid redundance. In addition, values smaller than 0.1 are not shown in the colormaps to highlight the trends that we will shortly discuss.

Figures 4.9a-c show the first-order contributions of μ, ξ, and v_0 to the variation of maximum flow height at each location. The mean values of $\hat{S}_\mu$, $\hat{S}_\xi$, and $\hat{S}_{v_0}$ over the 47958 locations are 0.3, 0.17, and 0.27 respectively. A closer look shows that the dry-Coulomb friction coefficient μ dominates in the downstream area beyond location B, whereas its impact in the upstream area of location B is limited; the turbulent

friction coefficient ξ is an influential factor in the upstream area of location B especially in areas around the major flow path, whereas it has negligible impact in the downstream area of location B; the release volume v_0 contributes the most in areas surrounding the release zone and has significant impact in areas near the minor flow path as well as areas surrounding location B, whereas it shows little influence in the downstream area similar as ξ.

Figures 4.9d-f present the first-order contributions of μ, ξ, and v_0 to the variation of maximum flow velocity at each location. The mean values of $\hat{S}_\mu$, $\hat{S}_\xi$, and $\hat{S}_{v_0}$ over all the locations are 0.34, 0.31, and 0.11 respectively. A closer inspection shows that the variation of maximum flow velocity in the downstream area beyond location B is predominantly driven by μ, while it has mild impact in the upstream area; ξ contributes the most to the variation of maximum flow velocity in the upstream area of location B, where the mean values of maximum flow velocity are large (comparing figure 4.9e with figure 4.8d); v_0 only has mild impact in areas near the release zone and near the minor flow path.

Comparing figures 4.9a-c with figures 4.9d-f, we find that the first-order contribution of μ to the variation of maximum flow height only slightly differs from its contribution to the variation of maximum flow velocity, with the mean over all locations increasing from 0.3 to 0.34. ξ has more impact on maximum flow velocity than on maximum flow height, with a difference 0.14 on average. The influence of v_0 on maximum flow height is more important than its influence on maximum flow velocity, with a difference 0.16 on average. The dominant role of μ in the downstream area agrees with the finding in section 4.5.1 that μ predominantly affects the angle of reach. The observation can be well explained based on Mangeney-Castelnau et al. (2003). More specifically, Mangeney-Castelnau et al. (2003) studied the forces involved in the momentum equation for the Coulomb friction law and found that the force caused by the dry-Coulomb friction is negligible in the early stage of the flow event (corresponds to the upstream area) while it becomes dominant in the later stage (corresponds to the downstream area). The importance of ξ in the upstream area with large mean values of maximum flow velocity is therefore expected since the turbulent friction term in equations 2.6-2.7 is proportional to the square of flow velocity and the role of the dry-Coulomb friction term is not important in this area. It should be noted that the turbulent term artificially limits the overestimated early stage velocity which results from the hydrostatic hypothesis used in depth-averaged shallow flow models, and therefore leads to more realistic early stage velocity (Garres-Díaz et al., 2021).

Figures 4.10a-c show the difference between total-effect and first-order Sobol' indices for maximum flow height at each location, which indicates the interactions between different input variables. Taking $\hat{S}_{T\mu} - \hat{S}_\mu$ as an example, it accounts for all high-order effects related to μ, including the second-order interaction between μ and ξ, the second-order interaction between μ and v_0, as well as the third-order interaction among μ, ξ, and v_0. The mean values of $\hat{S}_{T\mu} - \hat{S}_\mu$, $\hat{S}_{T\xi} - \hat{S}_\xi$, and $\hat{S}_{Tv_0} - \hat{S}_{v_0}$ over

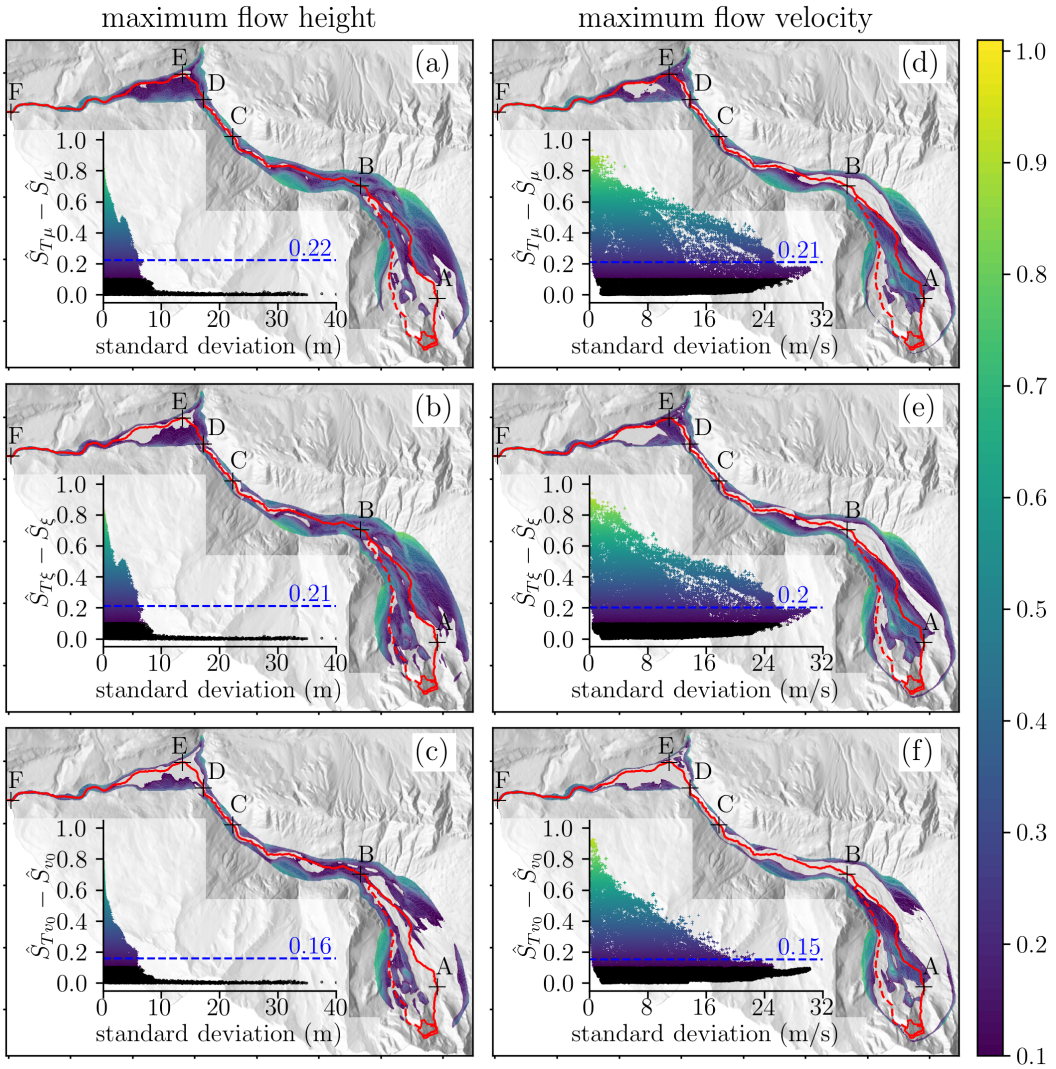

FIGURE 4.10: Difference between total-effect and first-order Sobol' indices for $(h_{l_1}^{\max}, \ldots, h_{l_k}^{\max})^T$ (left column) and for $(\|\mathbf{u}_{l_1}\|^{\max}, \ldots, \|\mathbf{u}_{l_k}\|^{\max})^T$ (right column). In each panel, values smaller than 0.1 are not shown in the colormap; the scatter plot shows the difference versus the standard deviation shown in figures 4.8b and 4.8e, where difference values larger than 0.1 are plotted using the same colorbar as that used for the colormap, and difference values smaller than 0.1 are plotted in black; the mean over all locations is notated in the scatter plot.

all locations are 0.22, 0.21, and 0.16 respectively. The areas where $\hat{S}_{T\mu} - \hat{S}_{\mu}$, $\hat{S}_{T\xi} - \hat{S}_{\xi}$, and $\hat{S}_{Tv_0} - \hat{S}_{v_0}$ have large values (see figures 4.10a-c) are generally in accord with the areas where the mean and standard deviation of maximum flow height have small values (see figures 4.8a-b), and the coefficient of variation of maximum flow height has large values (see figure 4.8c). One exception is the area around the major flow path between location A and location B. The value of $\hat{S}_{Tv_0} - \hat{S}_{v_0}$ in this exception area is very small (see figure 4.10c). It means that all high-order effects related to v_0 in this area are negligible, including the second-order v_0-μ interaction, the second-order v_0-ξ interaction, and the third-order v_0-μ-ξ interaction. The large values of $\hat{S}_{T\mu} - \hat{S}_{\mu}$ and $\hat{S}_{T\xi} - \hat{S}_{\xi}$ in this area as shown in figures 4.10a-b are therefore mainly due to

the second-order μ-ξ interaction since contributions from v_0-μ, v_0-ξ, and v_0-μ-ξ are negligible. From the inserted scatter plots in figures 4.10a-c which show respective difference versus the standard deviation, it is evident that the interactions generally decrease with increasing standard deviation. It means that the larger the variation of maximum flow height, the less the interactions between the three parameters.

Figures 4.10d-f show the difference between total-effect and first-order Sobol' indices for maximum flow velocity at each location. The mean values of $\hat{S}_{T\mu} - \hat{S}_{\mu}$, $\hat{S}_{T\xi} - \hat{S}_{\xi}$, and $\hat{S}_{Tv_0} - \hat{S}_{v_0}$ over all locations are 0.21, 0.2, and 0.15 respectively. Similar to the results on maximum flow height, the areas showing significant difference greatly resemble the areas with low mean values, low standard deviation values, and high coefficient of variation values of maximum flow velocity, see figures 4.8d-f. Again the area around the major flow path between location A and location B is an exception. It can be clearly seen from the scatter plots of respective difference versus the standard deviation, that the interactions generally decrease with increasing standard deviation.

Comparing figures 4.10a-c with figures 4.10d-f, the following trends can be observed for both maximum flow height and maximum flow velocity. First, most of the significant interactions occur on the margins of the flow paths where mean values and standard deviation values are relatively small, whereas values of coefficient of variation are relatively large (see figure 4.8). This may be due to the fact that a location on the margins is only reached by some of the forward simulations (hence some of the three parameter combinations). Second, the interactions generally decrease with increasing standard deviation. Third, there are stronger interactions between the two friction coefficients μ and ξ than between the release volume v_0 and each friction coefficient.

4.6 Summary

In this chapter, a computationally efficient approach which enables variance-based global sensitivity analyses of computationally demanding landslide run-out models has been presented. The methodology couples the novel open-source mass flow simulation tool r.avaflow (Mergili et al., 2017), robust Gaussian process emulation for multi-output models (Gu and Berger, 2016; Gu et al., 2018; Gu et al., 2019), and a recent algorithm addressing the emulator uncertainty (Le Gratiet et al., 2014). I have implemented a unified Python-based framework to seamlessly integrate r.avaflow, RobustGaSP, and SALib. Based on the 2017 Bondo landslide event, the approach is employed to study the global sensitivity of selected run-out model outputs to three input variables, namely the release volume and the two friction coefficients. The main findings are as follows.

- The proposed approach can be successfully used to study the relative importance and interactions of input variables in landslide run-out models, when

the trained Gaussian process emulators are validated and the base sample size of a Sobol' analysis is properly chosen.

- The first-order effects of each input variable are broadly in line with results of common one-at-a-time sensitivity analyses in the literature. The dry-Coulomb friction coefficient dominates the angle of reach, as well as maximum flow height and velocity in the downstream area. The turbulent friction coefficient contributes the most to the variation of maximum flow velocity in the area where maximum flow velocity values are expected to be large. The release volume is found to have significant impact on maximum flow height in the area surrounding the release zone whereas it shows little impact on maximum flow velocity.

- Interactions between the input variables could be analyzed for the full flow path, which cannot be assessed by commonly used one-at-a-time approaches. Significant interactions between the input variables generally happen on the margins of the flow path. The mean values and standard deviation values of maximum flow height and velocity are small in those areas. The interactions generally decrease with increasing variation of maximum flow height and velocity. Furthermore, there are stronger interactions between the two friction coefficients than between the release volume and each friction coefficient.

4.7 Modifications to the published paper

As mentioned in the outline (section 1.3), this chapter is a slightly modified version of the published paper: Hu Zhao, Florian Amann, Julia Kowalski. Emulator-based global sensitivity analysis for flow-like landslide run-out models, *Landslides*, 18, 3299-3314, 2021. HZ and JK conceived the idea. HZ developed the methodology, performed the implementation, case study, and result analyses with contributions from JK. FA provided data for the case study. HZ wrote the manuscript. JK and FA reviewed and revised the manuscript. Two anonymous reviewers reviewed the manuscript.

 The major modifications to the paper are listed below.

To avoid redundancies, following parts of the paper are left out in this chapter:

- The descriptions of landslide run-out models, uncertain factors, and surrogate modeling in the introduction of the published paper. They have been introduced in chapters 1-3 of this thesis.

- The mathematical details of landslide run-out modeling and Gaussian process emulation in section 2 of the published paper. They have been presented in chapter 2 and chapter 3 of this thesis.

- The recommendations for future work in the conclusion section of the published paper. They have been integrated with section 7.2 of this thesis.

Chapter 5

Bayesian active learning for parameter calibration of landslide run-out models

5.1 Literature review and motivation

As introduced in chapter 2, the rheological parameters in depth-averaged shallow flow type landslide run-out models mostly rely on calibration of past landslide events where field data are available. This is a typical inverse problem. Various calibration methods have been developed over past decades. They can be divided into two groups, namely deterministic and probabilistic methods, according to the form of calibration results. Deterministic methods aims to find the best-fit parameter configuration that leads to simulation outputs as close as possible to observed data. It is traditionally done by subjective trial-and-error calibration, such as in Hungr and McDougall (2009), Frank et al. (2015), and Schraml et al. (2015). More objective methods have been recently proposed by for example Calvello et al. (2017) and Aaron et al. (2019), where the best-fit parameter configuration is obtained by minimizing the mismatch between simulation outputs and observed data using optimization theory. There are two issues with deterministic methods: first, different parameter configurations may lead to similar simulation outputs, known as the non-uniqueness or equifinality problem (McMillan and Clark, 2009; Aaron et al., 2019); second, deterministic methods cannot account for measurement uncertainties.

Probabilistic methods, which avoid these two issues, aim to update prior knowledge of the calibration parameters to a posterior distribution based on observed data. They commonly require evaluating a run-out model at a large number of parameter configurations and depend on certain updating/selection rules. For example, Fischer et al. (2015) ran a depth-averaged flow model at 10000 rheological parameter points from a Latin hypercube design, obtained reduced parameter combinations based on a user-defined selection rule, and approximated posteriors of rheological parameters using a frequency analysis. Brezzi et al. (2016) obtained posteriors of rheological parameters by running 2000 simulations based on a Monte Carlo design and by applying the Kalman filter. Moretti et al. (2020) and Heredia et

al. (2020) approximated posteriors using 8000 and 50000 Markov chain Monte Carlo (MCMC) iterations within the Bayesian inference framework respectively. Aaron et al. (2019) approximated posteriors of rheological parameters using a full grid search within the Bayesian inference framework (Aaron, 2017). The main shortcoming of probabilistic methods is the high computational costs resulting from the large number of required simulation runs, as pointed out by many researchers (Fischer et al., 2015; Brezzi et al., 2016; Aaron, 2017; Heredia et al., 2020).

As a well-established surrogate modeling technique for reducing computational costs, Gaussian process (GP) emulation has been extensively used for parameter calibration in the past decades. One type of GP emulation-based strategy, pioneered by Kennedy and O'Hagan (2001), emulates input-output relations of the simulation model; see Bayarri et al. (2007), Higdon et al. (2008), Gu and Wang (2018). This type of method may need to emulate a simulation model with high-dimensional outputs if observed data are high-dimensional. An alternative type of GP emulation-based technique directly emulates the loss/likelihood function that measures the mismatch between simulation outputs and observed data. It avoids the high dimensional problem since the loss/likelihood function only has a scalar output. Examples are Oakley and Youngman (2017), Kandasamy et al. (2017), Fer et al. (2018), Wang and Li (2018), and Järvenpää et al. (2021). As for parameter calibration of landslide run-out models, Sun et al. (2021) built GP emulator for the landslide run-out model Massflow focusing on a scalar output (the run-out distance), and used the emulator for Bayesian inference of the model parameters. Navarro et al. (2018) used another surrogate modeling method, the polynomial chaos expansion, to approximate the landslide run-out model D-Claw, and used the surrogate for Bayesian inference of the model parameters. To my knowledge, no attempt has been made to directly emulate the loss/likelihood function regarding parameter calibration of landslide run-out models.

Another powerful technique to reduce computational costs is active learning. It has been recently used to improve inference quality. The essential idea is to iteratively run simulation at a new parameter point guided by all previous runs in order to increase our knowledge of the posterior the most (Cranmer et al., 2020). Various rules have been proposed regarding how a new parameter point may be selected. For instance, Zhang et al. (2016) adaptively constructed separate GP emulators for each model output, approximated the posterior using the MCMC method, and picked new parameter points by sampling from the approximated posterior. Kandasamy et al. (2017), Wang and Li (2018), and Järvenpää et al. (2021) sequentially constructed GP emulators for certain variations of the likelihood function (such as log-likelihood, log-unnormalized-posterior, etc.), and chose new parameter points that reduce the posterior uncertainty the most. To date, no active learning technique has been employed concerning parameter calibration of landslide run-out models.

As pointed out in the recent review by Cranmer et al. (2020), the rapidly advancing frontier of simulation-based inference driven by machine learning (here GP

emulation), active learning and a few other factors, is expected to profoundly impact many domains of science. Therefore, the main goal of this chapter is to develop an efficient parameter calibration method for landslide run-out modeling by integrating Bayesian inference, GP emulation, and active learning. A second goal is to study the effect of different observed data on calibration results using the new method.

5.2 Methodology

5.2.1 Bayesian inference framework

Bayesian inference of rheological parameters aims to derive a posterior probability distribution of the rheological parameters in light of observed data. The posterior distribution is computed using Bayes' theorem

$$p(\mathbf{x} \mid \mathbf{d}) = \frac{L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})}{\int L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})d\mathbf{x}}. \tag{5.1}$$

Here, $\mathbf{x}$ denotes the rheological parameters; $\mathbf{x} = (\mu, \xi)^T \in \mathbb{R}^2$ for the Voellmy rheological model. $\mathbf{d} = (d_1, \ldots, d_k)^T \in \mathbb{R}^k$ denotes observed data, such as the impact area, deposit area and volume, flow height and velocity at certain locations, etc. $p(\mathbf{x})$ denotes the prior distribution which encodes a priori knowledge about the rheological parameters before seeing the observed data. $L(\mathbf{x} \mid \mathbf{d}) = p(\mathbf{d} \mid \mathbf{x})$ is known as the likelihood, which is a function of $\mathbf{x}$. It measures how likely $\mathbf{x}$ takes certain values given the observed data. In the context of parameter calibration of a landslide run-out model, the likelihood $L(\mathbf{x} \mid \mathbf{d})$ involves the observed data $\mathbf{d}$ and corresponding outputs of the landslide run-out model, namely $\mathbf{y} = (y_1, \ldots, y_k)^T = \mathbf{f}(\mathbf{x})$. The exact form of the likelihood relies on the statistical ansatz used to model the residuals $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_k)^T$, where $\epsilon_i = d_i - y_i, i \in \{1, \ldots, k\}$. The residuals are commonly assumed to fulfill a k-variate Gaussian distribution with zero mean and a $k \times k$ covariance matrix $\boldsymbol{\Sigma}$, which leads to

$$L(\mathbf{x} \mid \mathbf{d}) = (2\pi)^{-\frac{k}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{d} - \mathbf{f}(\mathbf{x}))^T \boldsymbol{\Sigma}^{-1} (\mathbf{d} - \mathbf{f}(\mathbf{x}))\right\}. \tag{5.2}$$

The residuals $\epsilon_i, i \in \{1, \ldots, k\}$ are often further assumed to be independent, see for example Navarro et al. (2018), Aaron et al. (2019), and Moretti et al. (2020). In that case, the covariance matrix $\boldsymbol{\Sigma}$ reduces to a diagonal matrix

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{\epsilon_1}^2 & 0 & \ldots & 0 \\ 0 & \sigma_{\epsilon_2}^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & \sigma_{\epsilon_k}^2 \end{pmatrix}, \tag{5.3}$$

where σ_{ϵ_i} denotes the standard deviation of the residual $\epsilon_i, i \in \{1, \ldots, k\}$. The diagonal covariance matrix is used throughout this chapter. The proposed methodology can be easily extended to account for correlations without loss of generality.

The posterior $p(\mathbf{x} \mid \mathbf{d})$ in equation 5.1 cannot be computed in a closed form since a complicated landslide run-out model is involved in the likelihood. There exist various methods to approximate the posterior; see Gelman et al. (2013) for an overview. One type of method approximates the integral (the denominator in equation 5.1) using numerical integration. The term $L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})$ needs to be evaluated at a large number of points $\mathbf{x}_i, i \in \{1, \ldots, n\}$, which can be for example a set of evenly spaced full grid points. In terms of grid approximation, the posterior is in the form of

$$p(\mathbf{x} \mid \mathbf{d}) \approx \frac{L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})}{\sum_{i=1}^{n} L(\mathbf{x}_i \mid \mathbf{d})p(\mathbf{x}_i)\Delta\mathbf{x}_i}. \tag{5.4}$$

This type of method works well when $\mathbf{x}$ is low-dimensional, like two- or three-dimensional. When $\mathbf{x}$ is high-dimensional, MCMC methods are often used to draw a set of samples $\mathbf{x}_i, i \in \{1, \ldots, n\}$ based on the unnormalized posterior

$$p(\mathbf{x} \mid \mathbf{d}) \propto L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x}). \tag{5.5}$$

The posterior can then be estimated based on the MCMC samples by for example kernel density estimation.

No matter for which method, the unnormalized posterior $L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})$ needs to be evaluated at a large number of input points $\mathbf{x}_i, i \in \{1, \ldots, n\}$ in order to approximate the posterior with reasonable accuracy. Each evaluation requires running the landslide run-out model at $\mathbf{x}_i$. The computational cost can be prohibitively high if a single simulation takes a relatively long time. Similar to what has been done in chapter 4, one can build cheap-to-run emulator $\hat{\mathbf{f}}(\mathbf{x})$ to replace the expensive-to-run landslide run-out model $\mathbf{f}(\mathbf{x})$ in the likelihood function, and then perform Bayesian inference based on the emulator. It requires multivariate emulation if the observed data $\mathbf{d}$ are multi-dimensional (namely $k > 1$). An alternative approach is to directly emulate the logarithm of the unnormalized posterior. The logarithm form is commonly employed to avoid computational overflows and underflows (Gelman et al., 2013). Since the unnormalized posterior is a function of $\mathbf{x}$ with a scalar output, this type of method avoids the multivariate issue.

5.2.2 Gaussian process emulation for Bayesian inference

Let $g(\mathbf{x})$ denote the logarithm of the unnormalized posterior, namely

$$g(\mathbf{x}) = \ln\left[L(\mathbf{x} \mid \mathbf{d})p(\mathbf{x})\right]. \tag{5.6}$$

The idea here is to build a Gaussian process emulator $\hat{g}(\mathbf{x})$ for $g(\mathbf{x})$ based on training data $\{\mathbf{x}_i, g(\mathbf{x}_i)\}_{i=1}^{n^{tr}}$. Each $g(\mathbf{x}_i)$ requires running the landslide run-out model at input point $\mathbf{x}_i$. Once the GP emulator $\hat{g}(\mathbf{x})$ is obtained, the computationally expensive $g(\mathbf{x})$ in equation 5.1 can be replaced by $\hat{g}(\mathbf{x})$, which results in

$$p(\mathbf{x} \mid \mathbf{d}) \approx \frac{\exp(\hat{g}(\mathbf{x}))}{\int \exp(\hat{g}(\mathbf{x}))d\mathbf{x}}. \tag{5.7}$$

The posterior can then be estimated by applying the grid approximation or MCMC methods to equation 5.7. It reduces the number of required landslide run-out model

evaluations from n (usually thousands) to n^{tr} (a few hundreds), and therefore greatly improves the computational efficiency.

5.2.3 Active learning for training data design

The design of training data $\{\mathbf{x}_i, g(\mathbf{x}_i)\}_{i=1}^{n^{tr}}$ for emulator-based Bayesian inference deserves particular attention. Due to information gained from the observed data, the posterior is often localized in a small portion of the input space where the likelihood has large values, and is close to zero elsewhere. Since the aim of the inverse problem is to estimate the posterior reasonable well with limited computational cost, the commonly used space-filling sampling techniques, like the Latin hypercube design in section 4.4.3, are not efficient to determine training input points for the GP emulator $\hat{g}(\mathbf{x})$. More specifically, many input points from a space-filling sampling scheme will be located in areas where the posterior value is close to zero and therefore do not provide much information on the posterior that we want to estimate.

Active learning, also known as sequential design, is a simple but very impactful idea to wisely choose training input points at which the landslide run-out model needs to be run (Cranmer et al., 2020). Instead of choosing all the training input points a priori like in a space-filling sampling scheme, active learning iteratively chooses new training input point that is expected to increase our knowledge about the posterior the most. The selection of each new training input point is guided by all previous simulation runs.

Assume that b input points $\{\mathbf{x}_i\}_{i=1}^{b}$ have been chosen and $\{g(\mathbf{x}_i)\}_{i=1}^{b}$ have been correspondingly computed based on the b simulation runs and the observed data. Given $\{\mathbf{x}_i, g(\mathbf{x}_i)\}_{i=1}^{b}$, a GP emulator $\hat{g}_b(\mathbf{x})$ can be built using any method introduced in section 3.3. As for Bayesian inference with active learning, it is common to build the GP emulator using the maximum likelihood estimation (section 3.3.1) or marginal likelihood estimation (integrating out the regression parameters, section 3.3.2) in which the built emulator $\hat{g}_b(\mathbf{x})$ is still a Gaussian process; see for example Kandasamy et al. (2017), Sinsbeck and Nowak (2017), Wang and Li (2018), and Järvenpää et al. (2021). This choice is made mainly due to good properties of Gauss processes which we will shortly see. At any input point $\mathbf{x}^*$, the logarithm of the unnormalized posterior fulfills a Gaussian distribution, namely

$$\hat{g}_b(\mathbf{x}^*) \sim \mathcal{N}\left(m_b(\mathbf{x}^*), \sigma_b^2(\mathbf{x}^*)\right), \tag{5.8}$$

where the mean $m_b(\mathbf{x}^*)$ and variance $\sigma_b^2(\mathbf{x}^*)$ can be determined based on equations 3.11a-3.11c or equations 3.16a-3.16c.

The exponential term $\exp(\hat{g}_b(\mathbf{x}))$ gives an approximation of the unnormalized posterior. It encodes our current knowledge about the posterior and can therefore be used to determine the next input point that is expected to provide the most information on the posterior. A widely used strategy for active learning is to pick the input point at which the approximate unnormalized posterior $\exp(\hat{g}_b(\mathbf{x}))$ has the largest uncertainty. By running a new simulation at that point, the uncertainty at

Algorithm 2 Emulator-based Bayesian inference with active learning

1: Determine the number of total training input points n^{tr} and the number of initial training input points b_0.
2: Sample initial input points $\{\mathbf{x}_i\}_{i=1}^{b_0}$ using a Latin hypercube sampling scheme.
3: Run the landslide run-out model at each $\mathbf{x}_i$ and calculate $g(\mathbf{x}_i)$ based on the observed data and corresponding simulation outputs. Denote the training data as $\mathcal{D} = \{\mathbf{x}_i, g(\mathbf{x}_i)\}_{i=1}^{b_0}$
4: **for** $b = b_0, \ldots, (n^{tr} - 1)$ **do**
5: Build the GP emulator $\hat{g}_b(\mathbf{x})$ based on $\mathcal{D}$.
6: Determine the next input point $\mathbf{x}_{b+1}$ by maximizing equation 5.9a or 5.9b.
7: Run the landslide run-out model at $\mathbf{x}_{b+1}$ and calculate $g(\mathbf{x}_{b+1})$.
8: Augment training data $\mathcal{D} = \mathcal{D} \cup \{\mathbf{x}_{b+1}, g(\mathbf{x}_{b+1})\}$.
9: **end for**
10: Build the final GP emulator $\hat{g}_{n^{tr}}(\mathbf{x})$ based on $\mathcal{D} = \{\mathbf{x}_i, g(\mathbf{x}_i)\}_{i=1}^{n^{tr}}$.
11: Substitute $\hat{g}_{n^{tr}}(\mathbf{x})$ into equation 5.7 and estimate the posterior using grid approximation or MCMC methods.

that point will be eliminated and the gained information on the posterior is expected to be the most. Various uncertainty indicators have been proposed in the literature, such as the variance (Kandasamy et al., 2017) or entropy (Wang and Li, 2018) of the approximate unnormalized posterior $\exp(\hat{g}_b(\mathbf{x}))$. At any input point $\mathbf{x}^*$, $\exp(\hat{g}_b(\mathbf{x}^*))$ follows a log-normal distribution since $\hat{g}_b(\mathbf{x}^*)$ follows a Gaussian (normal) distribution. Both the variance $V_b(\mathbf{x}^*)$ and entropy $H_b(\mathbf{x}^*)$ of the log-normal distribution can be analytically computed, which have the forms of

$$V_b(\mathbf{x}^*) = \exp(2m_b(\mathbf{x}^*) + \sigma_b^2(\mathbf{x}^*))(\exp(\sigma_b^2(\mathbf{x}^*)) - 1), \tag{5.9a}$$

$$H_b(\mathbf{x}^*) = m_b(\mathbf{x}^*) + \frac{1}{2}\ln(2\pi e\sigma_b^2(\mathbf{x}^*)). \tag{5.9b}$$

The optimal input point $\mathbf{x}^*$ for next simulation run is given by the input point that maximizes equation 5.9a or equation 5.9b.

Algorithm 2 presents the proposed method for parameter calibration of landslide run-out models, which combines landslide run-out modeling, Bayesian inference, GP emulation, and active learning. A schematic illustration is given by figure 5.1. An initial GP emulator for the log-unnormalized posterior is first built based on initial b_0 simulation runs. Then the greedy one-step look ahead active learning strategy is employed to adaptively pick new input points, run simulations, and update training data . Last, the posterior distribution of the rheological parameters can be estimated based on the final GP emulator by using grid approximation or MCMC methods.

5.3 Implementation

The methodology proposed in section 5.2 consists of four components, including the simulator (Voellmy-type landslide run-out model; section 2.2), the emulator (univariate GP emulator; sections 3.2-3.3), the active learning strategy (section 5.2.3), and Bayesian inference (section 5.2.1). For each individual component, there are

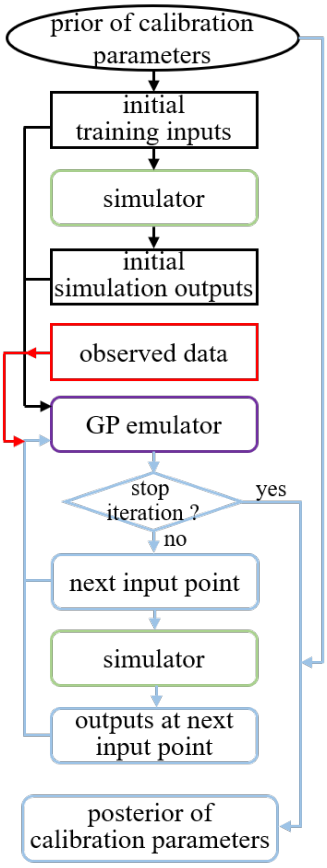

FIGURE 5.1: A schematic illustration of the emulator-based Bayesian active learning method.

some open-source software and/or packages available. However, as already mentioned in section 4.3, a unified framework that allows us to efficiently couple and leverage these building blocks is still lacking and therefore desirable. Our Python-based implementation provides such a framework. Among the four components, the simulator and emulator have already been implemented in a modular way based on the open-source software r.avaflow and R package RobustGaSP respectively; see section 4.3. They can be directly used in Algorithm 2. The rest two components are implemented as follows:

- **Active learning**. The key part of the active learning strategy is to solve the optimization problem at each iteration (Line 6 of Algorithm 2). It can be challenging if the uncertainty indicator has multiple local maxima (Wang and Li, 2018) and/or $\mathbf{x}$ is high-dimensional. As for rheological parameter calibration, $\mathbf{x}$ is generally low-dimensional (two-dimensional in terms of the Voellmy rheology). In that case, $\mathbf{x}_{b+1}$ can be determined using complete enumeration on a regular grid: first calculate the uncertainty indicator at each grid point $\mathbf{x}^*$ and then pick the grid point having the largest value of the uncertainty indicator as $\mathbf{x}_{b+1}$. This method has been used by for example Kandasamy et al. (2017) and Sinsbeck and Nowak (2017). It has been implemented in our Python-based framework without third-party software.

- **Bayesian inference**. The key idea of Bayesian inference is to update our prior knowledge of the calibration parameters to a posterior distribution given the observed data. As introduced in section 5.2.1, the posterior can be estimated using grid approximation or MCMC methods. The grid approximation is suitable for low-dimensional cases and MCMC methods can be used for both low- and high-dimensional cases. The grid approximation method (equation 5.4) is straightforward and has been implemented in our Python-based framework without third-party software. An advanced MCMC method, namely the affine-invariant ensemble sampler developed by Goodman and Weare (2010), is integrated into our framework by leveraging the Python package emcee (Foreman-Mackey et al., 2013).

5.4 Case study based on the 2017 Bondo landslide event

5.4.1 Types of observed data

Carrying out a parameter calibration requires the availability of observed data. Various types of observed data have been used in the literature, including the run-out distance (Sun et al., 2021), impact area, deposit distribution, deposit depth at specific locations, maximum flow velocity at specific locations (Aaron et al., 2019), time history of flow height at specific locations (Navarro et al., 2018), time history of the force exerted by flow mass onto the ground (Moretti et al., 2020), time history of flow velocity at the centre of flow mass (Heredia et al., 2020), etc. Among them, static data like the run-out distance, impact area, deposit distribution, and deposit height at specific locations can be obtained from pre- and post-event landscapes based on remote sensing, or post-event field investigation. The maximum flow velocity at certain locations can be obtained by for example post-event super-elevation measurements (Aaron et al., 2019). Dynamic data like time history of flow height or velocity are usually only available in lab or field experiments; see for example Navarro et al. (2018) and Heredia et al. (2020). The time history of the force exerted by flow mass onto the ground may be obtained based on seismic signals, which require records at seismic stations (Moretti et al., 2020).

While the methodology proposed in section 5.2 can be applied to any type of observed data and their combinations, the focus of this case study is put on parameter calibration using static data that are mostly available for real-world landslide events. More specifically, the impact area, deposit volume, deposit height at specific locations, and maximum flow velocity at specific locations will be used to calibrate the Voellmy rheological parameters based on a synthetic case.

5.4.2 Synthetic data generation

The case study in this chapter is based on the 2017 Bondo landslide event, which has been introduced in detail in section 4.4.1. The topography and distribution of the

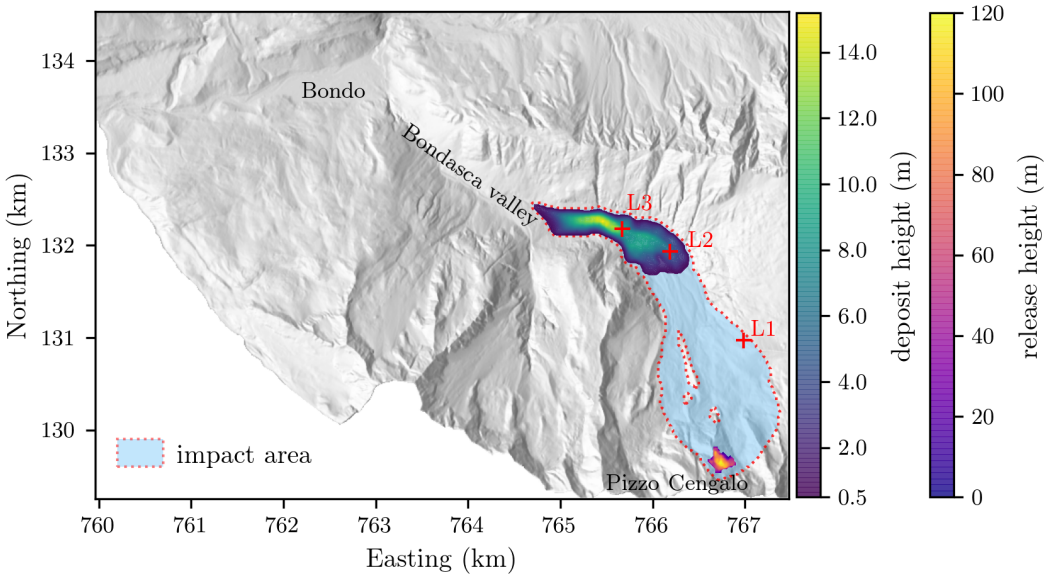

FIGURE 5.2: Topography and release mass distribution of the 2017 Bondo landslide event. The impact area and deposit distribution are simulation results with Voellmy rheological parameters $\mu = 0.23$ and $\xi = 1000\,\mathrm{m/s^2}$. The impact area and deposit area are determined using a cutoff value of 0.5 m.

release mass as shown in figure 4.2 are used here. In addition, the release volume is set to be 3 million m^3, which is equal to the release volume of the 2017 Bondo landslide event.

It should be noted that the purpose of the case study is not to calibrate the 2017 Bondo landslide event. Instead, the intention here is to test the proposed methodology and to study the impact of different types of observed data on calibration results. To this end, synthetic observed data are derived from given rheological parameters. Using these instead of field observations of the 2017 landslide event allows us to test the feasibility of the calibration in detail. The given rheological parameters serve as underlying truth which allows us to evaluate the proposed methodology.

The detailed procedure for generating synthetic observed data is as follows. First, a set of Voellmy rheological parameters, $\mu = 0.23$ and $\xi = 1000\,\mathrm{m/s^2}$, are picked from the ranges presented in section 4.4.2. Then the landslide run-out model is run given the rheological parameters, topography, and distribution of release mass. Next, the static data mentioned in section 5.4.1 are obtained by post-processing the simulation outputs. The impact area, deposit distribution, and locations where deposit height and/or maximum flow velocity are extracted, are noted in figure 5.2. Corresponding simulation results are summarized in table 5.1. Last, synthetic observed data are generated by adding random noises to the simulation results. For each synthetic observed data (each row in table 5.1 except the last two rows), the random noise is drawn from a Gaussian distribution with zero mean and an assumed standard deviation that equals 10% of the simulation result. Two additional synthetic observed data for the maximum flow velocity at L1 are generated using

TABLE 5.1: Simulation results ($\mu = 0.23$, $\xi = 1000\,\mathrm{m/s^2}$) and synthetic observed data.

Quantity of interest	Simulation result	Assumed standard deviation (σ_ϵ)	Random noise	Synthetic observed data
maximum velocity at L1 (m/s)	29.40	2.94	1.31	30.71
maximum velocity at L3 (m/s)	12.33	1.23	0.39	12.72
deposit height at L2 (m)	5.53	0.55	0.37	5.91
deposit height at L3 (m)	9.79	0.98	0.99	10.78
impact area (million m^2)	2.48	0.25	0.12	2.59
deposit volume (million m^3)	2.80	0.28	-0.09	2.71
maximum velocity at L1 (m/s)	29.40	5.88	2.62	32.03
	29.40	8.82	3.94	33.34

assumed standard deviations that equal to 20% and 30% of the simulation result respectively, as shown in the last two rows of table 5.1. They are used to investigate the impact of the uncertainty of observed data (section 2.4).

For future parameter calibration using real-world observed data, the standard deviations (σ_{ϵ_i}, $i \in \{1, \ldots, k\}$; equation 5.3) can be determined by heuristics when there are only coarse observed data (Aaron et al., 2019), or can be treated as calibration parameters in the Bayesian inference framework when there are rich observed data (Heredia et al., 2020).

5.4.3 Parameter calibration setup

The prior of the Voellmy rheological parameters is assumed to be a uniform distribution over the rectangular space defined by $\mu \in [0.02, 0.3]$ and $\xi \in [100, 2200]\,\mathrm{m/s^2}$. It means that no information about the rheological parameters, except the limiting values, is known before observing any data. This kind of prior is often used in the literature, such as Navarro et al. (2018), Aaron et al. (2019), and Moretti et al. (2020). More informative prior, such as the Gamma distribution, could also be used when expert knowledge and well known reference values are available (Heredia et al., 2020). A posterior obtained from previous calibration can also be used as the prior for a new calibration task when new observed data are available.

Given the synthetic observed data generated in section 5.4.2, the following cases are set up, as summarized in table 5.2. Note that L1, L2, and L3 refer to point locations as shown in figure 5.2. In cases 1-6, the Voellmy rheological parameters μ and ξ are calibrated based on a single synthetic observation using the proposed methodology. They are used to study the impact of different observed data on calibration results. The impact of the standard deviation σ_ϵ is investigated based on cases 1,

TABLE 5.2: Cases for rheological parameter calibration.

Number	Used synthetic observed data
1	maximum velocity at L1 ($\sigma_\epsilon = 2.94$)
2	maximum velocity at L3
3	deposit height at L2
4	deposit height at L3
5	impact area
6	deposit volume
7	maximum velocity at L1 ($\sigma_\epsilon = 5.88$)
8	maximum velocity at L1 ($\sigma_\epsilon = 8.82$)
9	maximum velocity at L1 ($\sigma_\epsilon = 2.94$) + deposit volume
10	maximum velocity at L3 + deposit height at L3

7 and 8. Cases 9 and 10 are designed to investigate the impact of combinations of observed data.

The number of initial training input points b_0 (step 1 in algorithm 2) is set as 40. Initial training input points $\{(\mu_i, \xi_i)^T\}_{i=1}^{40}$ are chosen using the maximin Latin hypercube design, as shown in figure 5.3. All the cases share the same 40 initial training input points, meaning that the initial 40 simulation runs only need to be conducted once. The number of total training input points n^{tr} is set to be 120, meaning that 80 adaptive simulation runs are used to actively learn the corresponding posterior in each case. Here, the active learning is performed based on equation 5.9b (step 6 of algorithm 2). It should be noted that the 80 adaptive simulation runs are distinct for each case since they are tailored to a specific observed data set in each case by active learning. It leads to in total 800 adaptive simulation runs for all the cases. For each case, the posterior is computed using grid approximation based on a 100×100 grid (step 11 of algorithm 2).

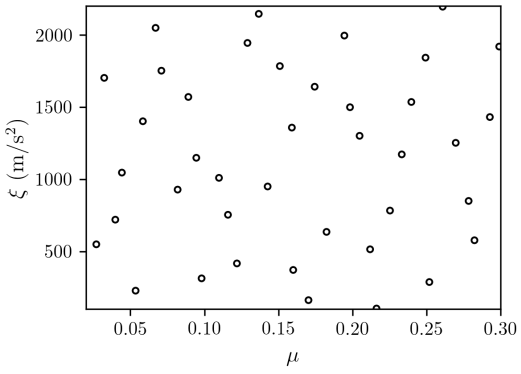

FIGURE 5.3: Initial training input points sampled using the maximin Latin hypercube design.

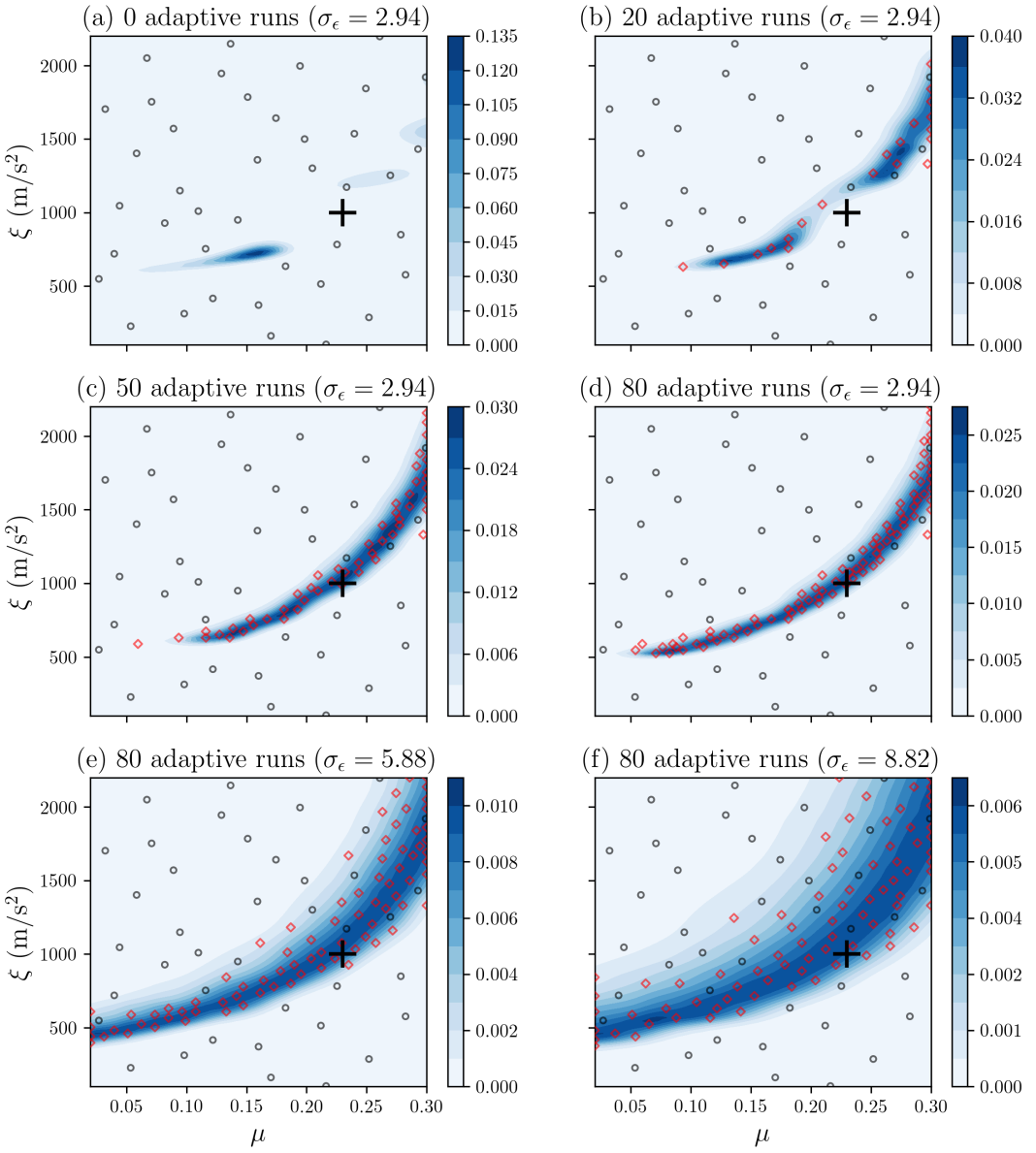

FIGURE 5.4: Estimated posteriors based on synthetic maximum velocity at location L1: (a)-(d) case 1 ($\sigma_\epsilon = 2.94$; see table 5.2) at 0, 20, 50, 80 adaptive simulation runs, respectively; (e) case 7 ($\sigma_\epsilon = 5.88$) at 80 adaptive simulation runs; (f) case 8 ($\sigma_\epsilon = 8.82$) at 80 adaptive simulation runs. In each panel, the black cross shows the underlying true values of μ and ξ, which are 0.23 and 1000 m/s^2; the black circles denote the 40 initial training input points; the red diamonds represent the input points that are adaptively determined by active learning; the colormap shows the posterior of the rheological parameters which is estimated based on the initial and adaptive training runs.

5.5 Results and discussions

5.5.1 Active learning process and the impact of σ_ϵ

Figures 5.4a-d show the estimated posteriors at four different iteration numbers in case 1 (see table 5.2). They provide a direct impression on how the active learning works and the impact of the number of iteration steps on the calibration result. For

each given iteration number i, a GP emulator $\hat{g}_{40+i}(\mathbf{x})$ is built based on the simulation runs up to the given iteration, namely 40 initial runs and i adaptive runs. Then the posterior at that iteration number can be computed by substituting $\hat{g}_{40+i}(\mathbf{x})$ into equation 5.7 and applying grid approximation (see step 11 of algorithm 2). From figures 5.4a-d, the following observations can be made:

- The quality of the estimated posterior is quite poor without any adaptive simulation run as expected. It gets gradually improved with increasing iteration number, owing to the information gain from simulation runs at adaptively chosen input points.
- The underlying truth of the rheological parameters locates very close to the high probability regions of the final posterior (figure 5.4d). It implies that the proposed methodology is capable of correctly calibrating the rheological parameters.
- The final posterior has high values not only in regions near the underlying truth, but also in regions far from the underlying truth. It highlights the non-uniqueness or equifinality problem associated with deterministic calibration methods as introduced in section 5.1. Namely different configurations of the rheological parameters may lead to similar simulation outputs. Probabilistic calibration methods should be therefore used whenever possible.
- The final posterior occupies only a small portion of the parameter space and has values close to zero elsewhere. In this case, the active learning scheme which adaptively determines input points has great advantage, since it allocates more computational resources for exploring the high probability regions. In other words, the active learning scheme can provide better approximation of the posterior than a pure space filling design scheme with the same computational budget.

Figures 5.4e and 5.4f show the final posteriors of case 7 and case 8 (based on synthetic maximum velocity at location L1 with $\sigma_\epsilon = 5.88$ and $\sigma_\epsilon = 8.82$ respectively). Comparing them with figure 5.4d, it can be seen that the underlying truth of the rheological parameters is still close to the high probability regions, but the shape of the posterior becomes flat with the increase of the standard deviation σ_ϵ. This result is expected and is similar to the findings of Aaron et al. (2019) and Sun et al. (2021). More specifically, increasing σ_ϵ means increasing uncertainty of the observed data. The information gained from the observed data (encoded in the likelihood function) therefore decreases with increasing σ_ϵ. The posterior accordingly relies more on the prior information, here a uniform distribution.

In order to investigate the convergence behaviour of the active learning scheme, the change of the total variation distance with respect to the number of iterations for cases 1, 7, and 8 is plotted in figure 5.5. The total variation distance measures

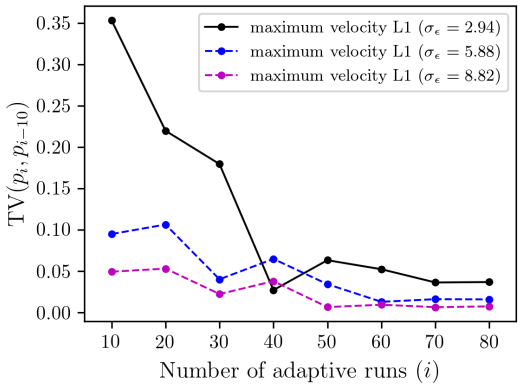

FIGURE 5.5: The change of the total variation distance with respect to the number of adaptive runs for cases 1, 7, and 8 (see table 5.2). After each 10 adaptive runs, the total variance distance between p_i and p_{i-10} is calculated, where p_i and p_{i-10} denote the estimated posterior based on simulation runs up to the i-th and $(i-10)$-th iteration respectively.

the difference between two probability density distributions $p(\mathbf{x})$ and $p'(\mathbf{x})$, and is defined as (Järvenpää et al., 2021)

$$\mathrm{TV}(p, p') = \frac{1}{2} \int |p(\mathbf{x}) - p'(\mathbf{x})| d\mathbf{x}. \tag{5.10}$$

In each case, the total variation distance $\mathrm{TV}(p_i, p_{i-10})$ is iteratively calculated after each 10 adaptive runs, where p_i and p_{i-10} denote the estimated posterior based on simulation runs up to the i-th and $(i-10)$-th iteration respectively. It can be seen from figure 5.5 that $\mathrm{TV}(p_i, p_{i-10})$ for all three cases generally decreases with increasing number of adaptive runs and remains at a relatively low value after certain number of adaptive runs. It implies that the estimated posterior for each case reaches a stable stage. It should be noted that the total variation distance, or other quantities that measure the difference between two probability distributions like the Kullback-Leider divergence, can be used to design early stopping criteria for algorithm 2, see for example Wang and Li (2018).

Based on above results, it can be concluded that the proposed emulator-based Bayesian active learning method is able to correctly calibrate the rheological parameters. Compared to commonly used probabilistic methods without emulation techniques as mentioned in section 5.1, the proposed method greatly improves the computational efficiency by reducing the number of necessary simulation runs from thousands (even tens of thousands) to a few hundreds. Compared to emulator-based Bayesian inference without active learning, the proposed method can provide better approximation of the posterior if the computational budget is the same by wisely allocating computational resources.

5.5.2 Different observed data and their combinations

Figures 5.6a-f show the final posteriors for cases 1-6 respectively. The underlying truth of the rheological parameters and 80 adaptively chosen input points are also plotted in the figures. The following observations can be made:

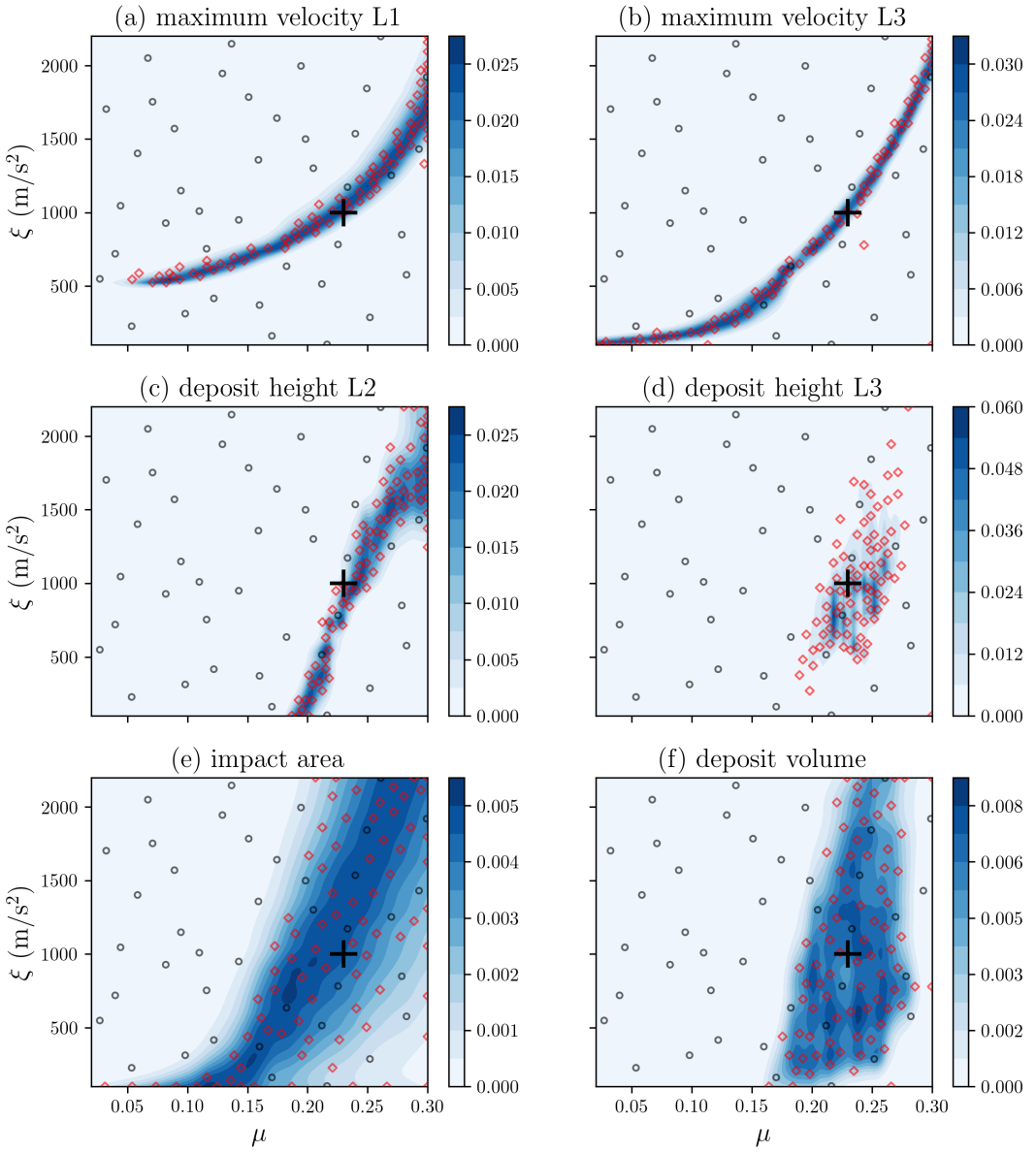

FIGURE 5.6: (a)-(f) estimated posteriors at 80 adaptive simulation runs for cases 1-6 (see table 5.2). In each panel, the black cross shows the underlying true values of μ and ξ; the black circles denote the 40 initial training input points; the red diamonds represent the 80 input points that are adaptively determined by active learning; the colormap shows the posterior of the rheological parameters which is estimated based on the 40 initial and 80 adaptive simulation runs.

- The underlying truth of the rheological parameters closely locates near the high probability regions, no matter which synthetic observed data are used for parameter calibration. It further validates that the proposed methodology is able to correctly calibrate the rheological parameters.

- The posteriors obtained based on different synthetic observed data significantly differ from one another. It means that the information of the rheological parameters gained from different types of observed data can be greatly different.

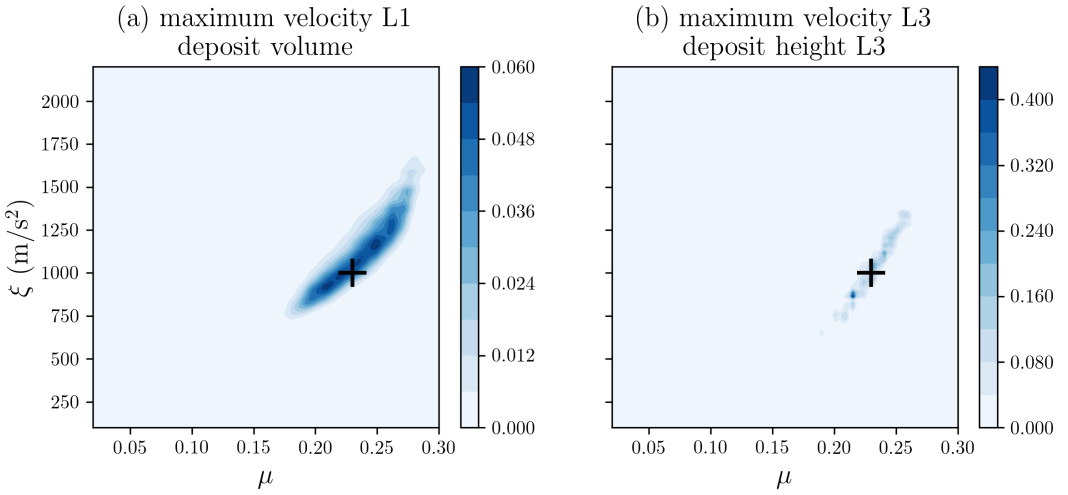

FIGURE 5.7: Estimated posterior at 80 adaptive simulation runs for (a) case 9 and (b) case 10.

- Location-wise observed data, such as maximum velocity and deposit height at specified locations, can better constrain the rheological parameters than aggregated overall observed data, like the impact area and deposit volume.
- A single observation alone is not enough to constrain the rheological parameters. It implies that different types of observed data should be combined in order to effectively calibrate the rheological parameters.

Based on the results shown in figures 5.6a-f, it can be presumed that in this case the rheological parameters can be better constrained if the calibration relies on a combination of complementary observed data, such as maximum velocity at L1 and deposit volume or maximum velocity at L3 and deposit height at L3. In order to validate the presumption, the rheological parameters are calibrated using the combination of maximum velocity at L1 and deposit volume, and the combination of maximum velocity and deposit height at L3, respectively. The results are shown in figures 5.7a-b. It can be seen that the resulted posteriors are better constrained in each case as expected. In the future, such an analysis can serve as a starting point to optimize data acquisition, e.g. which sensors to use and where to place them, such that we can maximize the knowledge return of field observations.

5.6 Summary

In this chapter, an efficient probabilistic parameter calibration method for landslide run-out models is developed by integrating Bayesian inference, GP emulation, and active learning. The methodology couples the novel open-source mass flow simulation tool r.avaflow, R package RobustGaSP for robust Gaussian process emulation, and an active learning scheme in the Bayesian inference framework. All these components have been implemented in the unified Python-based framework. The feasibility and efficiency of the proposed method are investigated based on a case study

with synthetic observed data. The impact of different types of observed data is also studied using the proposed method. The main findings are as follows:

- The proposed emulator-based Bayesian active learning method is able to correctly calibrate the rheological parameters. Compared to commonly used probabilistic methods without emulation techniques, the proposed method greatly improves the computational efficiency by reducing the number of necessary simulation runs from thousands (even tens of thousands) to a few hundreds. Compared to emulator-based Bayesian inference without active learning, the proposed method can provide better approximation of the posterior if the computational budget is the same.
- The information gained from different types of observed data can greatly differ from one another. Location-wise observed data like maximum velocity and deposit height at specific locations provide better constraint for the Voellmy rheological parameters than aggregated overall observed data like the impact area and deposit volume. In addition, a single observation alone cannot effectively constrain the Voellmy rheological parameters. Different types of observed data should be combined in order to improve the quality of the posterior.

Chapter 6

Monte Carlo-type uncertainty quantification of landslide run-out models

Since the inputs of landslide run-out models involve large uncertainties (sections 2.3-2.4), it is essential to quantity the impact of the uncertainties on model outputs. Among the various uncertainties, the uncertainty of the initial mass and the uncertainty of rheological parameters are low-dimensional. It means that their uncertainties can be represented by a small number of random variables. For example, the uncertainty of the initial mass can be parameterized as a random variable determining the release volume and a few random variables determining the coordinates of vertexes of the release polygon. Monte Carlo-type uncertainty quantification of these low-dimensional inputs is straightforward: first, a number of random input points are drawn from the distributions of these input factors; then the landslide run-out model is run at each of the random input point; last, the uncertainty of a model output is evaluated by its low-order moments like the mean and variance, or its complete probability distribution. A Gaussian process emulator can be straightforwardly built to replace the landslide run-out model in order to improve computational efficiency. Figure 6.1 shows a schematic workflow of emulator-based Monte Carlo-type uncertainty quantification of those low-dimensional uncertain factors. The results shown in figure 4.8 can be viewed as an emulator-based Monte Carlo-type uncertainty quantification of the release volume and the Voellmy rheological parameters (assumed to be uniformly distributed), where the concerned model outputs are point-wise maximum flow height and velocity.

In contrast, the uncertainty of topographic data is high-dimensional and more difficult to deal with. Since topographic uncertainty is often neglected in current practice, this chapter mainly focuses on topographic uncertainty quantification. Extension to other uncertain factors is however straightforward.

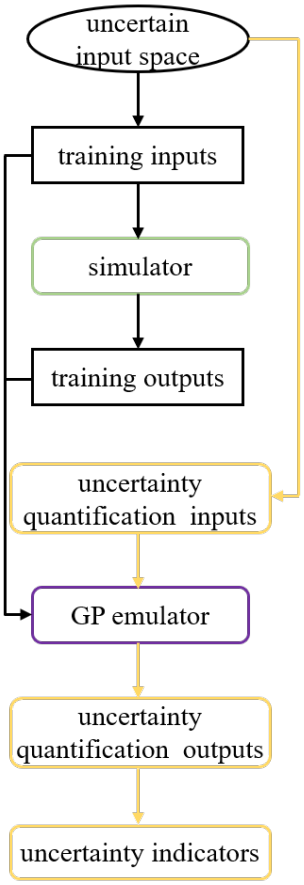

FIGURE 6.1: A schematic representation of the emulator-based Monte Carlo-type uncertainty quantification method. It can be directly applied to low-dimensional uncertain factors like the initial condition of flow mass and rheological parameters. Its extension to high-dimensional topographic uncertainty will be discussed in section 6.6.

6.1 Literature review and motivation

Topographic data are an indispensable input to any of the computational landslide run-out models introduced in section 2.1. They represent the terrain in which the slide is likely to occur and determine the components of the gravitational acceleration in the governing equations 2.1-2.3. Pioneered by Miller and Laflamme (1958), digital elevation models (DEMs) have become the most popular form of representing topographies in the scientific community. Methods for generating DEMs have evolved rapidly over decades from conventional approaches, like field surveying and topographic-map digitizing, to passive and active remote sensing, such as stereoscopic photogrammetry, interferometric synthetic aperture radar (InSAR), and light detection and ranging (lidar); see Wilson (2012) for a comprehensive review. Differences between these methods exist in terms of their footprint, cost, resolution, and accuracy of the resulting DEM. Whatever method used, however, the resulting DEM will inevitably contain errors that are introduced during source data acquisition and/or during data processing. The so-called DEM error hence refers to the difference between the true real-world elevations and their DEM representation.

Typically, there is a lack of information on the DEM error, which has led to the notion of "DEM uncertainty". It refers to what we do not know about the error (Wechsler, 2007).

Nowadays, several global DEM databases, e.g., SRTM (Rodriguez et al., 2006), AW3D30 (Courty et al., 2019), and TanDEM-X (Wessel et al., 2018), as well as some regional DEM databases (Pakoksung and Takagi, 2016) are publicly available. Commercial DEM databases that are associated with significant costs also exist (Hawker et al., 2018). Online initiatives such as OpenTopography facilitate community access and aim to democratize online availability of high-resolution topography data acquired with lidar and other technologies (Krishnan et al., 2011). Despite a broad variety of existing DEM sources, however, we are still facing (and will continue to face in the near future) a very limited availability of high-accuracy DEMs for some regions that are particularly prone to landslide hazards, e.g., in Asia (Froude and Petley, 2018). Whenever using DEM data for simulation-based landslide hazard analysis, it is hence important to be aware of DEM error and uncertainty and to consider their potential impact on computational run-out analyses and related computational risk assessments.

DEM error has had the attention of researchers for a long time. Many efforts have for instance been put into quantifying the error associated with specific DEM sources based on data of higher accuracy, e.g., acquired by satellite measurements (Berry et al., 2007; Mouratidis and Ampatzidis, 2019), medium-footprint lidar (Hofton et al., 2006), or GPS survey (Rodriguez et al., 2006; Bolkas et al., 2016; Patel et al., 2016; Wessel et al., 2018; Elkhrachy, 2018). Meanwhile, a variety of methods have been devised to classify DEM error into various categories (Oksanen, 2003; Hengl et al., 2004; Fisher and Tate, 2006). Due to the complexity of potential influencing factors (sensor technology, retrieval algorithms, data processing, land cover and surface morphology, terrain attributes; Wilson, 2012; Fisher and Tate, 2006; Gonga-Saholiariliva et al., 2011), these methods can only constrain the DEM error and will not deterministically correct for it at all grid points. Hence, DEM uncertainty remains and has to be accounted for in any subsequent analysis that relies on the DEM data.

In this circumstance, a Monte Carlo-type stochastic simulation is an effective computational approach to deal with the situation (Holmes et al., 2000). Instead of considering a single (assumed as accurate) DEM, the fundamental idea of a stochastic simulation in the context of DEM uncertainty propagation is to separate the DEM into a known deterministic contribution and an unknown DEM error. DEM uncertainty is then accounted for by treating the DEM error as a random field consisting of a collection of random variables defined at selected grid points. An ensemble of equiprobable realizations of the random field is then generated based on certain assumptions and available information about DEM error. This could for instance be the so-called root mean square error (RMSE), a minimalist indicator for the overall error magnitude, or a semivariogram that provides information about the spatial autocorrelation of the DEM error. Adding the DEM error realizations to the known

deterministic DEM contribution results in an ensemble of equiprobable DEM realizations, which can finally be used for a DEM uncertainty propagation analysis.

Stochastic simulation methods for DEM uncertainty propagation analyses have been developed since the 1990s and are by now widely applied in many fields, including terrain analysis (Holmes et al., 2000; Raaflaub and Collins, 2006; Moawad and EI Aziz, 2018), flood modeling (Watson et al., 2015; Hawker et al., 2018; Kiczko and Miroslaw-Swiatek, 2018), soil erosion modeling (Aziz et al., 2012), landslide susceptibility mapping (Qin et al., 2013), and dry-block-and-ash-flow modeling (Stefanescu et al., 2012). With respect to rapid, flow-like landslide run-out modeling, very little work has been done to assess the potential impact of DEM uncertainty, most likely due to the complexity, and hence level of sophistication, of the associated process models. It is therefore desirable to consider the impact of DEM uncertainty on landslide run-out models, as overlooking DEM uncertainty may lead to a bias in risk management decisions.

The goal of this chapter is threefold. The first is to introduce two types of geostatistical methods (unconditional and conditional stochastic simulations) that allow us to account for DEM uncertainty in landslide run-out analyses (sections 6.2-6.3). The second goal is to study whether DEM uncertainty is critical to landslide run-out modeling and affects the modeling's results based on a case study using the two approaches (sections 6.4-6.5). The last goal is to propose a new approach which enables efficient uncertainty quantification (especially topographic uncertainty quantification) of landslide run-out models (section 6.6).

6.2 Methodology

The topographic surface can be expressed as a function $z(\mathbf{s})$ in the Cartesian coordinate system $\{x, y, z\}$ (see figure 2.1), where $\mathbf{s} = (s_x, s_y)^T$ denotes horizontal coordinates of a location. In practice, the function $z(\mathbf{s})$ is often constructed from discrete gridded raster data. Discretizing a domain of interest $\mathcal{D}$ into the horizontal x and y direction results in a spatial grid

$$\mathbf{S}_{mn} = \{\mathbf{s}_{ij} = (s_{xi}, s_{yj}) \mid (s_{xi}, s_{yj}) \in \mathcal{D};\ i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n\}. \tag{6.1}$$

The elevation data associated with each grid point $\mathbf{s}_{ij}$ are

$$\mathbf{Z}_{mn} = \{z_{ij} = z(\mathbf{s}_{ij}) \mid \forall\ \mathbf{s}_{ij} \in \mathbf{S}_{mn}\}. \tag{6.2}$$

The elevation $\mathbf{Z}_{mn}$ of a common DEM data product might be erroneous with respect to the true values as discussed in section 6.1. Denoting the true elevation as

$$\mathbf{Z}^{\star}_{mn} = \{z^{\star}_{ij} = z^{\star}(\mathbf{s}_{ij}) \mid \forall\ \mathbf{s}_{ij} \in \mathbf{S}_{mn}\}, \tag{6.3}$$

the DEM error can be expressed as

$$\boldsymbol{\mathcal{E}}_{mn} = \{\varepsilon_{ij} = z^{\star}_{ij} - z_{ij} \mid \forall\ \mathbf{s}_{ij} \in \mathbf{S}_{mn}\}. \tag{6.4}$$

If the error $\boldsymbol{\mathcal{E}}_{mn}$ was known, the real-world topographic surface $\mathbf{Z}^{\star}_{mn}$ would be recovered. The fact, however, that the error is unknown or only limited information about it is available implies an uncertainty in the topographic input to the landslide run-out models. Within this thesis, the uncertainty associated with the unknown DEM error is referred to as DEM uncertainty. In this circumstance, each ε_{ij} is treated as a random variable and $\boldsymbol{\mathcal{E}}_{mn}$ is accordingly treated as a random field, which consists of a collection of random variables ε_{ij}. By generating multiple realizations of the random field $\boldsymbol{\mathcal{E}}_{mn}$, DEM uncertainty can be represented. This process is widely known as stochastic simulation. It requires a suitable model to describe the jointed uncertainty in all ε_{ij} based on limited available information about DEM error. The task can be further divided into determining (1) the probability distribution function (pdf) of each ε_{ij} which quantifies local uncertainty at each grid point and (2) the correlation between different ε_{ij} which is usually termed spatial autocorrelation of DEM error.

According to available information on the DEM error, existing approaches that could be used to solve the two issues can be generally classified into two groups:

- unconditional stochastic simulation (USS),
- conditional stochastic simulation (CSS).

More specifically, USS is only informed of properties of DEM error, e.g., the RMSE, and thus does not honor any actual DEM error values. CSS is informed of a certain number of actual DEM error values at reference locations within the DEM, e.g., obtained from higher-accuracy reference data, and thus could directly honor the actual DEM error values at reference locations (Fisher and Tate, 2006).

6.2.1 Unconditional stochastic simulation

Typically available information about the DEM error provided by DEM vendors is the root mean square error (RMSE). For a set of k reference locations, it is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(z_i^{\star} - z_i)^2}. \tag{6.5}$$

Here, $z_i^{\star} := z^{\star}(\mathbf{s}_i)$ denotes a higher-accuracy elevation value at a reference location $\mathbf{s}_i \in \mathcal{D}$ and $z_i := z(\mathbf{s}_i)$ denotes the corresponding elevation value based on the DEM. The higher-accuracy elevation values and corresponding elevation values based on the DEM at k reference locations are collectively denoted as $\mathbf{z}_k^{\star}$ and $\mathbf{z}_k$ respectively.

It should be noted that, while the RMSE is typically available, this is not true for the reference elevation values $\mathbf{z}_k^{\star}$ themselves. As stated numerous times in the literature, it is critical that the RMSE only provides a global indication of DEM error magnitude without any information about its spatial autocorrelation. Still, it is by far the most widely used DEM error indicator for many DEM databases and mostly

the only available information that is included with DEM products. In this circumstance, USS could be used to represent DEM uncertainty and study its propagation into landslide run-out analyses.

Modeling DEM uncertainty based on USS assumes that all local error values ε_{ij} are independent and fulfill the same univariate Gaussian distribution with a mean (μ_ε) of zero and a standard deviation (σ_ε) equivalent to the given RMSE. Under this assumption, an ensemble of spatially uncorrelated realizations of the random field $\boldsymbol{\mathcal{E}}_{mn}$ can be generated by randomly assigning error values to each ε_{ij} according to its local Gaussian probability distribution.

In the next step, the (unknown) spatial autocorrelation of $\boldsymbol{\mathcal{E}}_{mn}$ has to be taken into account. Potential methods that could be applied are simulated annealing, spatial autoregressive modeling, spatial moving averages, etc.; see Wechsler (2007). Simulated annealing is generally computationally intensive, and spatial autoregressive modeling becomes impractical for the simulation of large areas (Oksanen, 2006). In this chapter, the spatial moving-averages method is used. It increases the spatial autocorrelation by filtering spatially uncorrelated realizations with a distance-weighted filter proposed by Wechsler and Kroll (2006). For ε_{ij} at one grid point of an uncorrelated realization, its value is replaced by the weighted average of ε_{ij} at all grid points within the filter kernel. The weight decreases with increasing distance to the grid point, which is similar to semivariogram trends (Wechsler and Kroll, 2006). The size of the filter denoted as d depends on the maximum autocorrelation length of $\boldsymbol{\mathcal{E}}_{mn}$ which again is unknown if the RMSE is the only available information. In practice, d is often determined based on the maximum autocorrelation length of the original DEM (Wechsler, 2007; Aziz et al., 2012).

By applying the spatial moving-averages technique, the ensemble of spatially uncorrelated realizations of the random field $\boldsymbol{\mathcal{E}}_{mn}$ turns into spatially correlated realizations. Adding these DEM error realizations to the deterministic DEM leads to an ensemble of equiprobable DEM realizations. Given each of the DEM realizations as topographic input, the landslide run-out model can be run and interested outputs can be extracted. The impact of topographic uncertainty on interested outputs can then be quantified by statistically analyzing the ensemble of interested outputs. Algorithm 3 summarizes the steps of using unconditional stochastic simulation for topographic uncertainty quantification.

Though it relies on some assumptions, such as an appropriate choice of correlation length d, the sketched approach is generally applicable if the RMSE is the only available information. It may become critical if a DEM contains a systematic bias which means that the mean of $\mathbf{z}_k^\star - \mathbf{z}_k$ deviates largely from zero. More specifically, if following Fisher and Tate (2006) and Wessel et al. (2018) in defining mean μ_ε and standard deviation σ_ε as

$$\mu_\varepsilon = \frac{1}{k}\sum_{i=1}^{k}(z_i^\star - z_i), \quad \sigma_\varepsilon = \sqrt{\frac{1}{k-1}\sum_{i=1}^{k}((z_i^\star - z_i) - \mu_\varepsilon)^2}, \tag{6.6}$$

Algorithm 3 USS for topographic uncertainty quantification

1: Determine the RMSE, correlation length d, and the number of realizations N.
2: **for** $n = 1, \ldots, N$ **do**
3: Sample a spatially uncorrelated realization of the random field $\boldsymbol{\mathcal{E}}_{mn}$ by drawing $m \times n$ points from a univariate Gaussian distribution with zero mean and a standard deviation equivalent to the given RMSE.
4: Obtain a spatially correlated realization of $\boldsymbol{\mathcal{E}}_{mn}$ by applying the spatial moving-averages technique with the correlation length d to the uncorrelated realization.
5: Obtain a DEM realization by adding the spatially correlated realization of $\boldsymbol{\mathcal{E}}_{mn}$ to the deterministic DEM.
6: Run the landslide run-out model given the DEM realization as topographic input.
7: Extract interested model outputs.
8: **end for**
9: **return** Interested model outputs of the N landslide run-out model evaluations.
10: Perform statistical analysis on the model outputs to quantify the impact of topographic uncertainty.

the RMSE can be expressed in terms of μ_ε and σ_ε as

$$\text{RMSE} = \sqrt{{\mu_\varepsilon}^2 + \frac{k-1}{k}\sigma_\varepsilon^2}. \tag{6.7}$$

If the number of reference points k is relatively large, $\sqrt{(k-1)/k}$ is close to 1. Equation 6.7 then indicates that the RMSE is larger than the standard deviation σ_ε if the mean μ_ε deviates from zero. The difference between the RMSE and σ_ε increases with increasing μ_ε. For example, the μ_ε, σ_ε, and RMSE of the global TanDEM-X DEM based on about 3 million reference points are 0.17, 1.28, and 1.29 m (Wessel et al., 2018). Those of the EU-DEM of Central Macedonia based on 12 943 reference points are 1.8, 3.6, and 4.0 m, while those of the ASTER GDEM of the same area based on the same reference points are 6.8, 7.6, and 10.2 m (Mouratidis and Ampatzidis, 2019). This means that assuming the standard deviation of the DEM error to be given as the RMSE consequently overestimates the variability in the DEM error if the mean deviates largely from zero.

The implications of both issues, namely the fact that the filter size d is unknown and has to be subjectively chosen and that the RMSE provides an insufficient representation of the DEM error, are investigated in the following study. For convenience, the two issues are referred to as

- unrepresentative RMSE and
- subjective d.

The USS method greatly resembles the Gaussian process prior defined in equations 3.1-3.4. More specifically, the mean in the USS method is assumed to be zero which can be achieved by setting the regression parameters in equation 3.2 to zero; the square of the RMSE plays the same role as the variance σ^2 in equation 3.3;

the spatial moving-averages technique has the same functionality as the correlation function in equation 3.3 and the correlation length d is simply the correlation length parameters $\boldsymbol{\psi}$ reduced to an isotropic case. The above two issues of USS can therefore be understood in the context of GP regression as follows: due to the lack of training data (actual DEM error values at reference locations), the variance σ_ε^2 and correlation length d of the DEM error can only be approximated by an unrepresentative RMSE and a subjectively chosen value of d.

6.2.2 Conditional stochastic simulation

Higher-accuracy reference data at certain reference locations $\mathbf{s}_1, \ldots, \mathbf{s}_k$ are sometimes available, e.g., from higher-accuracy DEM products or GPS surveys. Note that, although these data might be subject to error themselves, it is fair to assume this error to be much smaller. This justifies the use of the higher-accuracy reference data as true elevation values $\mathbf{z}_k^\star$. Given $\mathbf{z}_k^\star$, the DEM error $\varepsilon_1^\star, \ldots, \varepsilon_k^\star$ at the k reference locations can be obtained, which are collectively denoted as $\boldsymbol{\varepsilon}_k^\star$. In this case, conditional stochastic simulation can be used to generate realizations of the random field $\boldsymbol{\mathcal{E}}_{mn}$. Many geostatistical methods of conditional simulation could be applied, including sequential simulation algorithms, the p-field approach, and simulated annealing (Goovaerts, 1997). In this chapter, the sequential Gaussian simulation method is employed. It is the most attractive technique for stochastic spatial simulation according to Temme et al. (2009) and has been widely utilized in DEM uncertainty propagation analysis (Holmes et al., 2000; Aziz et al., 2012). The sequential Gaussian simulation method can be viewed as a combination of kriging and a sequential sampling algorithm.

Kriging is the name of Gaussian process regression in geostatistics (section 3.1). Typical kriging equations are given in equations 3.11a-3.11c. In the context of DEM uncertainty, the training data $\mathbf{y}^{tr}$ in equations 3.11a-3.11c refer to the known DEM error values at the reference locations $\boldsymbol{\varepsilon}_k^\star$; the output y^* at an untried input $\mathbf{x}^*$ refers to the DEM error at an unobserved location. A realization of the random field $\boldsymbol{\mathcal{E}}_{mn}$ can therefore be directly generated by sampling from the kriging equations (Gaussian process posterior) using the matrix decomposition-based technique (equation 3.24).

As already mentioned in section 3.4, the main problem of the sampling technique based on matrix decomposition is its computational complexity. Regarding generating realizations of $\boldsymbol{\mathcal{E}}_{mn}$, the computational complexity is $\mathcal{O}\left((m \times n)^3\right)$. It may cause problems when $m \times n$ is large (like hundreds of thousands), which is often the case for large scale DEMs. The sequential sampling algorithm is therefore used to overcome this issue. Instead of directly sampling a realization of $\boldsymbol{\mathcal{E}}_{mn}$, it sequentially samples each ε_{ij} along a random path which contains all grid points $\mathbf{s}_{ij}$, $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$. According to the kriging equations 3.11a-3.11c, each ε_{ij} fulfills a univariate Gaussian distribution denoted as $\mathcal{N}(\mu_{\varepsilon_{ij}}, \sigma_{\varepsilon_{ij}}^2)$. The essential idea now is that the mean $\mu_{\varepsilon_{ij}}$ and standard deviation $\sigma_{\varepsilon_{ij}}$ are determined sequentially, namely not only conditioned on $\boldsymbol{\varepsilon}_k^\star$ but also on previously sampled ε_{ij}.

Algorithm 4 CSS for topographic uncertainty quantification

1: Determine a semivariogram model to represent the spatial autocorrelation of DEM error based on normal score transformed $\boldsymbol{\varepsilon}_k^\star$, and determine the number of realizations N.
2: **for** $n = 1, \ldots, N$ **do**
3: Define a random path visiting each location $\mathbf{s}_{ij}$ once.
4: Sample each ε_{ij} sequentially along the random path; at each location, ε_{ij} is sampled from the kriging equations that are based on the semivariogram model, normal score transformed $\boldsymbol{\varepsilon}_k^\star$, and samples of ε_{ij} at previously visited locations.
5: Obtain a realization of $\boldsymbol{\mathcal{E}}_{mn}$ by back-transforming all sampled ε_{ij} to the original distribution of $\boldsymbol{\varepsilon}_k^\star$.
6: Obtain a DEM realization by adding the realization of $\boldsymbol{\mathcal{E}}_{mn}$ to the deterministic DEM.
7: Run the landslide run-out model given the DEM realization as topographic input.
8: Extract interested model outputs.
9: **end for**
10: **return** Interested model outputs of the N landslide run-out model evaluations.
11: Perform statistical analysis on the model outputs to quantify the impact of topographic uncertainty.

It is noteworthy that the variogram or semivariogram is often used in the kriging procedure in the geostatistical literature, instead of the covariance function. The covariance function $C(\cdot, \cdot)$ is often known as the covariogram in geostatistics. In the context of DEM uncertainty, the covariogram defines the covariance between DEM error values at any two locations $\mathbf{s}_i$ and $\mathbf{s}_j$. Under the assumption of stationary (section 3.2.1), the covariogram is given by

$$C(\varepsilon(\mathbf{s}_i), \varepsilon(\mathbf{s}_j)) = C(\mathbf{s}_i - \mathbf{s}_j) = C(\mathbf{h}), \tag{6.8}$$

where $\mathbf{h} \in \mathbb{R}^2$ denotes the difference between $\mathbf{s}_i$ and $\mathbf{s}_j$. The variogram is related to the covariogram as follows:

$$V(\varepsilon(\mathbf{s}_i), \varepsilon(\mathbf{s}_j)) = V(\mathbf{h}) = V(\mathbf{s}_i) + V(\mathbf{s}_j) - 2C(\mathbf{h}) = 2C(0) - 2C(\mathbf{h}). \tag{6.9}$$

The semivariogram is simple half the variogram, namely

$$\gamma(\varepsilon(\mathbf{s}_i), \varepsilon(\mathbf{s}_j)) = \gamma(\mathbf{h}) = C(0) - C(\mathbf{h}). \tag{6.10}$$

Algorithm 4 summarizes the steps of using sequential Gaussian simulation (conditional stochastic simulation) for topographic uncertainty quantification. The normal score transform of $\boldsymbol{\varepsilon}_k^\star$ (the known DEM error values at reference locations) is applied to fulfill the format requirements of Gaussian simulation (Holmes et al., 2000).

6.3 Implementation

The methodology presented in section 6.2 implies the following implementation work: first, the unconditional and conditional stochastic simulation methods need

to be implemented in order to generate a number of DEM realizations to represent DEM uncertainty. Second, a landslide run-out model solver is required to run simulations given the DEM realizations. Third, the extensive output data need to be postprocessed. In addition, understanding how DEM uncertainty affects terrain attributes may facilitate the interpretation of its impact on landslide run-out modeling. This requires the ability to calculate terrain attributes, e.g., slope and roughness, of the original DEM as well as of the generated DEM realizations. To solve these tasks, a workflow that integrates my own Python implementation and existing software is developed. The main modules of the workflow are as follows:

- **DEM uncertainty representation**. This module is used to generate an ensemble of N equally probable DEM realizations to represent DEM uncertainty based on available information about DEM error. USS as introduced in section 6.2.1 is implemented without third-party software for cases in which only the RMSE is available. For cases in which higher-accuracy reference data are provided, CSS as introduced in section 6.2.2 is implemented by integrating the sequential Gaussian simulation algorithm of the Stanford Geostatistical Modeling Software (SGeMS; Remy et al., 2009) into the workflow.

- **Simulator**. This module is used to run landslide run-out simulations given the DEM realizations. Christen et al. (2010) presented the proprietary mass flow simulation platform RAMMS which provides a GIS-integrated implementation of the landslide run-out model employing the Voellmy rheology (section 2.2). I implemented a Python-based wrapper for RAMMS to automatically set up required simulation inputs, run simulations, and extract outputs of interest.

- **Statistical analysis and visualization**. This module is used to conduct statistical analysis on the outputs of interest and visualize results. It is mainly based on the Python NumPy and Matplotlib modules. For example, a probabilistic hazard map can be produced to indicate potential hazard area.

- **Terrain analysis.** This module is used to analyze terrain characteristics of the original DEM and DEM realizations, which may help us to interpret the outputs of interest. This is achieved by integrating several terrain analysis tools from WhiteboxTools (Lindsay, 2018) like calculating slope, aspect, roughness index, etc., into our workflow.

The above RAMMS-based simulator (running on WINDOWS operating systems) can be simply replaced by the r.avaflow-based simulator (running on LINUX operating systems) that has been used in chapter 4 and chapter 5, and vice versa. The implementation of both simulators therefore allows utilizing the unified framework developed in this thesis on both WINDOWS and LINUX operating systems.

It should be noted that due to the high dimensionality of DEM uncertainty (each ε_{ij} is a random variable), it is difficult to directly employ GP emulation to emulate the

relations between DEM uncertainty and outputs of interest. For the small-scale case studied in sections 6.4-6.5, it is not an issue since the simulation can be run relatively fast (around one minute). For computationally expensive cases like the Bondo case in chapter 4 and chapter 5, it may become an issue. In section 6.6, a new approach based on dimensionality reduction of DEM uncertainty is proposed. It allows us to represent DEM uncertainty by a limited number of random variables and therefore makes GP emulation applicable.

6.4 Case study based on the 2008 Yu Tung landslide event

The case study is based upon a historic landslide and two DEM sources. For the purpose of DEM uncertainty propagation analysis, one DEM source is assumed to be more accurate than the other. Higher-accuracy reference data are then obtained from the more accurate DEM source to assess elevation error in the less accurate DEM source. A series of computational scenarios are designed based on the higher-accuracy reference data to study the impact of DEM uncertainty on landslide run-out simulation for both the case where only the RMSE is available and the case where higher-accuracy reference data are available. Additional computational scenarios are designed to study the unrepresentative RMSE and subjective d issues as detailed in section 6.2.1.

6.4.1 Case background and DEM sources

A historic landslide happened on 7 June 2008 on the hillside above Yu Tung Road in Hong Kong due to an intense rainfall event; see figure 6.2. It was the largest flow-like landslide out of 19 landslides during that event. Around 3400 m^3 of material was mobilized and traveled about 600 m until it was deposited. The landslide event had a severe infrastructural impact, as it led to closure of the westbound lanes of Yu Tung Road for more than 2 months (AECOM Asia Company Limited, 2012). The Yu

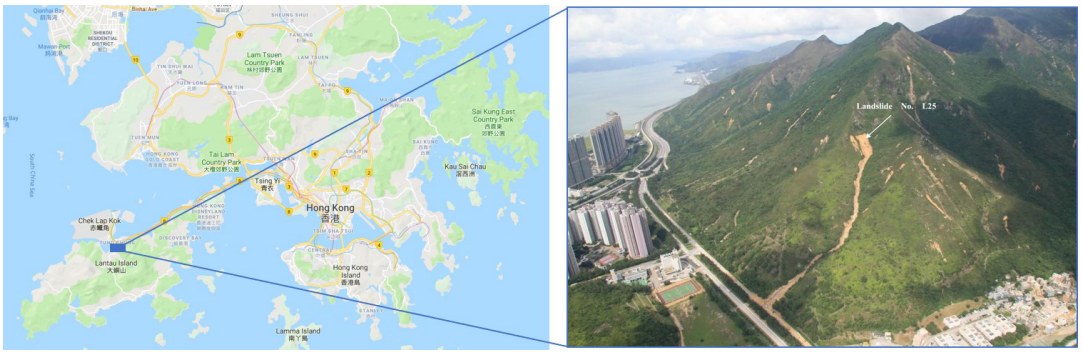

FIGURE 6.2: The 2008 Yu Tung Road landslide. Left: Google map of Hong Kong (map data © 2019). Right: aerial photograph of Yu Tung Road site after the 2008 landslide. It corresponds to the No. L25 landslide in the GEO Report (AECOM Asia Company Limited, 2012).

Tung Road landslide also served as a benchmark case for predictive landslide run-out analysis at the second JTC1 workshop on triggering and propagation of rapid flow-like landslides in Hong Kong 2018 (Pastor et al., 2018). Two types of DEM data of the Yu Tung Road area were the basis for this case study:

- A public 5 m resolution digital terrain model covering the whole area of Hong Kong (HK-DEM). It can be downloaded from the website of the Survey and Mapping Office of Hong Kong. The HK-DEM is generated from a series of digital orthophotos, which are derived from aerial photographs taken in 2014 and 2015. The reported accuracy is ± 5 m at a 90% confidence level. (DATA.GOV.HK, 2019)

- A 2 m resolution DEM (2m-DEM) covering the main area of the Yu Tung Road landslide event. Its boundary is shown in figure 6.3a. It was provided for the benchmark exercise during the second JTC1 workshop. It is produced based on the field mapping after the 2008 Yu Tung Road landslide event and a pre-event DEM. According to the "note to participants" of the second JTC1 workshop (which can be found under the link http://www.hkges.org/JTC1_2nd/be.html, last access: 18 May 2020), the 2m-DEM represents the rupture surface in the release zone area and the pre-event slope surface in other areas. The rupture surface is obtained based on the field mapping (AECOM Asia Company Limited, 2012).

The 2m-DEM is assumed to be more accurate than the 5 m resolution HK-DEM. Similar to the consideration at the beginning of section 6.2.2, the 2m-DEM and 5 m resolution HK-DEM correspond to $\mathbf{Z}^\star$ and $\mathbf{Z}$ as defined in section 6.2. A set of higher-accuracy reference data $\mathbf{z}_k^\star$ can hence be determined to provide information to represent uncertainty in the 5 m resolution HK-DEM. At the channel base, information on vegetation in the 2m-DEM is not available. The 5 m resolution HK-DEM includes vegetation. Any vegetation present in the channel base in the 5 m resolution HK-DEM is not explicitly accounted for in the sense of a modeled DEM correction. Rather, it is subsumed as part of the DEM error.

It should be noted that, due to a different time of DEM data acquisition, there are infrastructural factors present in the 5 m resolution HK-DEM but not in the 2m-DEM. After the time of data acquisition of the 2m-DEM, debris-resisting barriers and a road were built in the area within the red ellipse and blue rectangle, respectively, in figure 6.3a. They are reflected in the HK-DEM but not in the 2m-DEM, which leads to large inconsistency between the two DEMs in that area. Therefore, to avoid an unrealistically large error in the HK-DEM, data from the 2m-DEM in that area are excluded from higher-accuracy reference data $\mathbf{z}_k^\star$.

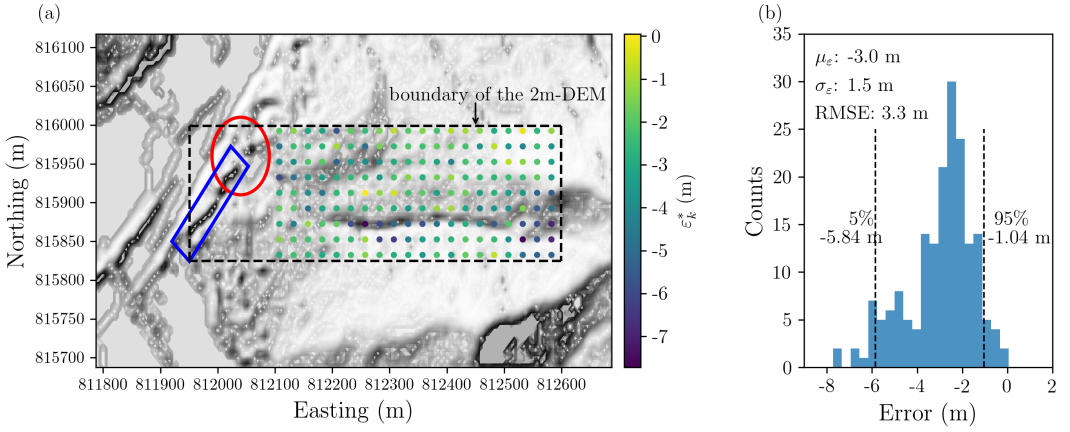

FIGURE 6.3: (a) Elevation error $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$ of the HK-DEM at 180 reference locations. The background is the hillshade plot of the HK-DEM. Debris-resisting barriers and a road in the area indicated by the red ellipse and blue rectangle constructed after the 2008 landslide event are represented in the HK-DEM but not in the 2m-DEM. It causes inconsistency between the two DEMs in that area. To avoid an unrealistically large error in the HK-DEM, data from the 2m-DEM in that area are excluded from higher-accuracy reference data. (b) Histogram of $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$.

6.4.2 DEM realizations

6.4.2.1 Information on DEM error

As shown in Fig. 6.3a, 180 evenly spaced reference locations are picked from the HK-DEM grid points within the boundary of the 2m-DEM. Higher-accuracy reference data at these locations are obtained from the 2m-DEM using bilinear interpolation, denoted as $\mathbf{z}_k^\star\{k = 180\}$. By subtracting the corresponding elevation values of the HK-DEM $\mathbf{z}_k\{k = 180\}$ from $\mathbf{z}_k^\star\{k = 180\}$, elevation error values of the HK-DEM at the 180 reference locations are obtained, denoted as $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$.

Figure 6.3b shows the histogram of $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$. Of the elevation error values, 90% are within -5.84 m and -1.04 m, which is close to the reported accuracy (see section 6.4.1). The μ_ε, σ_ε, and RMSE according to equation 6.6 and equation 6.5 are -3.0, 1.5, and 3.3 m, respectively. Here, it should be noted that the RMSE is larger than the σ_ε since the μ_ε is not zero, which indicates a systematic bias. As discussed in section 6.2.1, this also indicates that assuming the standard deviation of the HK-DEM error to be equivalent to the RMSE in USS would overestimate the variability in the HK-DEM error.

Based on $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$, an isotropic semivariogram model can be determined. It describes the spatial autocorrelation of the HK-DEM error and has a form of

$$\gamma(\mathbf{h}) = 0.1 \times \mathrm{Sph}\left(\frac{||\mathbf{h}||}{180}\right) + 0.9 \times \mathrm{Exp}\left(\frac{||\mathbf{h}||}{50}\right). \tag{6.11}$$

Here, $\mathrm{Sph}(\cdot)$ and $\mathrm{Exp}(\cdot)$ denote the basic spherical and exponential semivariogram models (Goovaerts, 1997); $||\mathbf{h}||$ denotes the horizontal distance between any two locations. A comparison between the experimental semivariance values based on $\boldsymbol{\varepsilon}_k^\star\{k = 180\}$ and the parameterized semivariogram model given by equation 6.11

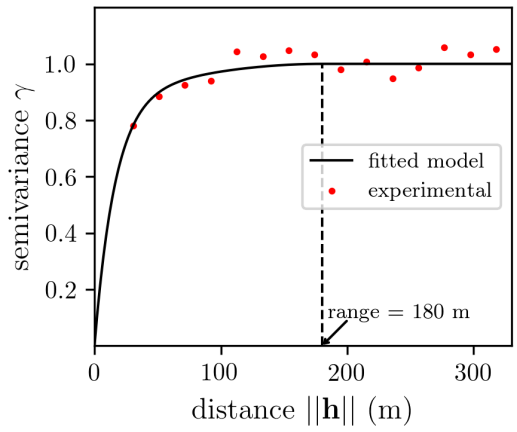

FIGURE 6.4: Experimental semivariances based on $\varepsilon_k^\star\{k = 180\}$ and fitted parameterized semivariogram model given by equation 6.11. The range of the semivariogram model is 180 m.

can be seen in figure 6.4. The continuous semivariogram model is fitted to the experimental semivariance values so as to deduce semivariance values for any possible distance $||\mathbf{h}||$ required by simple kriging (Goovaerts, 1997). The range of the semivariogram model is 180 m. It indicates the maximum autocorrelation length of the HK-DEM error, on which the size of the spatial moving filter d depends (see section 6.2.1).

6.4.2.2 DEM uncertainty scenarios

As mentioned in section 6.2, DEM users are often restricted to DEM error information in the form of a single RMSE value per data product. Rarely, they have higher-accuracy reference data. In order to account for both situations, two corresponding information levels are considered in the following study.

a. *Rudimentary error information – the RMSE only.* In this situation, the RMSE is assumed to be the only available error information for the 5 m resolution HK-DEM. In order to compare results to (b), we employ the RMSE 3.3 m as generated based on $\mathbf{z}_k^\star\{k = 180\}$ as well as the size of the spatial moving filter d of 180 m to match the range of the fitted semivariogram model in figure 6.4. USS introduced in section 6.2.1 is used to generate N realizations of the HK-DEM, denoted as $\mathrm{USS}_N\{\mathrm{RMSE} = 3.3, d = 180\}$.

b. *Highly informed – higher-accuracy reference data.* In this situation, $\mathbf{z}_k^\star\{k = 180\}$ is assumed to be available. That means the error $\varepsilon_k^\star\{k = 180\}$ at the reference locations are exactly known, as well as the fitted semivariogram model based on $\varepsilon_k^\star\{k = 180\}$. CSS introduced in section 6.2.2 is used to generate N realizations of the HK-DEM, denoted as CSS_N.

Following the two nominal scenarios (a) and (b) that are based on specific error $\varepsilon_k^\star\{k = 180\}$ at reference locations determined from the available data sources, it is

also useful to analyze the impact of unrepresentative RMSE and subjective d issues of USS (section 6.2.1). This could help to understand to what extent the results of USS can be trusted, if only a single RMSE value per data product is available. Three additional values of the RMSE that are 0.5, 1.5, and 2.5 m with a fixed d of 180 m are used as inputs for USS to study the unrepresentative RMSE issue. It should be noted that the RMSE 1.5 m corresponds to the true standard deviation σ_ε based on $\varepsilon_k^\star\{k = 180\}$ (see figure 6.3b). Another three additional values of d that are 0, 90, and 270 m with a fixed RMSE of 3.3 m are used to consider the subjective d issue. The corresponding realizations of the HK-DEM are denoted as $\mathrm{USS}_N\{\mathrm{RMSE} = 0.5, 1.5, 2.5, d = 180\}$ and $\mathrm{USS}_N\{\mathrm{RMSE} = 3.3, d = 0, 90, 270\}$. To sum up, all the scenarios for stochastic simulation are listed in table 6.1.

TABLE 6.1: Scenarios for stochastic simulation.

	Method to generate DEM realizations	Input to generate DEM realizations
Scenario (a)	USS	RMSE $= 3.3$; $d = 180$
Scenario (b)	CSS	semivariogram; $\varepsilon_k^\star\{k = 180\}$
Scenarios for unrepresentative RMSE	USS	RMSE $= 0.5, 1.5, 2.5$; $d = 180$
Scenarios for subject d	USS	RMSE $= 3.3$; $d = 0, 90, 270$

6.4.2.3 Number of DEM realizations

The integrity of a stochastic simulation requires a large number of DEM realizations, and more realizations naturally require many computational resources. Thus one has to find a reasonable compromise. Typically, this can be found through a representative convergence study. Since the aim is to study the impact of topographic uncertainty on landslide run-out simulation, the relative change in topographic attributes with an increasing number of HK-DEM realizations is analyzed. Here, 1000 HK-DEM realizations are generated for the two information levels (a) and (b) as introduced in section 6.4.2.2, namely $\mathrm{USS}_{N=1000}\{\mathrm{RMSE} = 3.3, d = 180\}$ and $\mathrm{CSS}_{N=1000}$, respectively. Topographic attributes including slope, aspect, and roughness at all HK-DEM grid points are calculated for each realization.

An indicator of the relative change that is similar to Raaflaub and Collins (2006) is defined to investigate the converging behavior. Taking slope as an example, for a given number N of HK-DEM realizations, the standard deviation of slope at each HK-DEM grid point over the N realizations is calculated. The calculated standard deviation values at all grid points constitute a grid of standard deviation values. Then the standard deviation of the grid of standard deviation values is calculated, which leads to a single standard deviation value for the given number N. For each N from 1 to 1000, a standard deviation value can be correspondingly calculate. The same procedure is applied to aspect, roughness, and elevation.

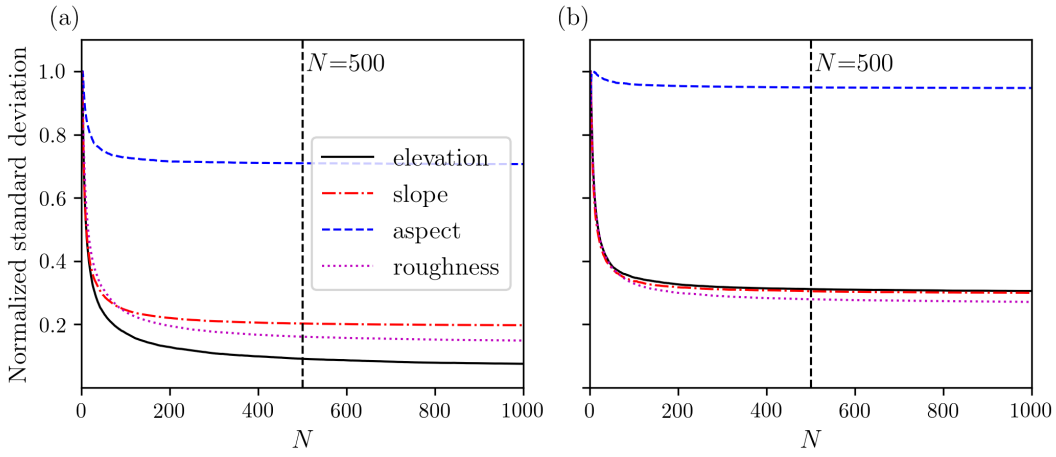

FIGURE 6.5: The relative change in topographic attributes with respect to the number of HK-DEM realizations. The realizations are generated with (a) $\text{USS}_N\{\text{RMSE} = 3.3, d = 180\}$; (b) CSS_N. Beyond $N = 500$, adding more realizations has little impact on topographic attributes.

Figure 6.5 shows plots of normalized standard deviation of the grid of standard deviation values with respect to the number of HK-DEM realizations for the two situations (a) and (b). It can be seen that for situation (a), aspect levels out first, followed by slope, roughness, and elevation. Beyond 500 realizations, there is little change in normalized standard deviations. This indicates that adding more realizations has little impact on topographic attributes. For situation (b), aspect also levels out first while the other three attributes show less difference. Compared to (a), (b) converges faster, which indicates CSS requires fewer DEM realizations than USS does. Nevertheless, $N = 500$ is set for the remainder of this chapter both for USS and CSS. Namely, 500 HK-DEM realizations are generated for each scenario input set as listed in table 6.1.

6.4.2.4 Statistics of DEM error realizations

In order to conduct a further quality check of the implementation of both USS and CSS, corresponding DEM error realizations of the $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ and $\text{CSS}_{N=500}$ scenarios are investigated. The DEM error realizations are denoted as $\text{USS}^{\text{Error}}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ and $\text{CSS}^{\text{Error}}_{N=500}$, respectively. Ideally, the local mean $\mu_{\varepsilon_{ij}}$ and standard deviation $\sigma_{\varepsilon_{ij}}$ of DEM error realizations at each grid point $\mathbf{s}_{ij}$ should match the underlying assumptions as introduced in section 6.2 if the number of DEM error realizations is sufficiently large.

Figures 6.6a and 6.6c show the mean and standard deviation grid of the $\text{USS}^{\text{Error}}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$. It can be seen that the mean values at all grid points are centered around 0 m. The standard deviation values are centered around 3.3 m. This corresponds to the assumption underlying USS that all ε_{ij} fulfill the same univariate Gaussian distribution with a mean (μ_ε) of zero and a standard deviation (σ_ε) given by the RMSE (see section 6.2.1).

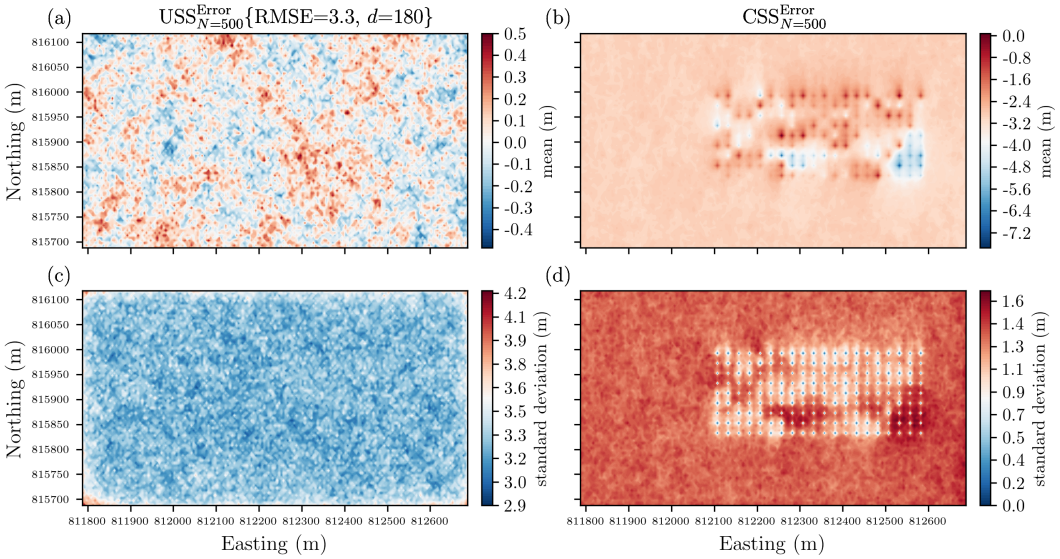

FIGURE 6.6: Statistics of HK-DEM error realizations. (a) Mean and (c) standard deviation grid of $USS^{Error}_{N=500}\{RMSE = 3.3, d = 180\}$. (b) Mean and (d) standard deviation grid of $CSS^{Error}_{N=500}$.

Figures 6.6b and 6.6d show the mean and standard deviation grid of the $CSS^{Error}_{N=500}$. The mean values at grid points away from the reference locations are centered around the mean (μ_ε) -3.0 m based on $\varepsilon^\star_k\{k = 180\}$. They become close to $\varepsilon^\star_k\{k = 180\}$ with the decrease in distance between grid points and the reference locations and are equal to $\varepsilon^\star_k\{k = 180\}$ at the reference locations. Similarly, the standard deviation values at grid points away from the reference locations are centered around the standard deviation (σ_ε) 1.5 m based on $\varepsilon^\star_k\{k = 180\}$. They vanish at the reference locations. This also corresponds to the assumption underlying CSS that each ε_{ij} fulfills a univariate Gaussian distribution with a mean $\mu_{\varepsilon_{ij}}$ and standard deviation $\sigma_{\varepsilon_{ij}}$ given by the kriging mean and standard deviation at $\mathbf{s}_{ij}$ (see section 6.2.2).

6.4.3 Landslide run-out simulation setup

As introduced in section 2.3, initial distribution of the flow mass, topographic data (DEM), and rheological parameters are key inputs for performing a deterministic landslide run-out simulation. For all scenarios, the release area as provided for the benchmark exercise during the second JTC1 workshop is consistently used. It matches the release area of the 2008 Yu Tung Road landslide (Pastor et al., 2018) as shown in figure 6.7b. The release height is assumed to be 1.2 m, leading to a release volume of around 2900 m^3, based on the 5 m resolution HK-DEM. The Voellmy rheological parameters μ and ξ are set to be 0.105 and 300 m/s^2, respectively. They are suggested in the GEO Report issued by the Civil Engineering and Development Department of Hong Kong, which are obtained using back analysis with information from a video capturing the lower portion of the landslide and detailed field mapping after the landslide (AECOM Asia Company Limited, 2012). The HK-DEM and

all HK-DEM realizations generated in section 6.4.2 are used as topographic inputs.

Let's call a deterministic landslide process simulation based on a DEM as a simulation run and N deterministic landslide process simulations based on N DEM realizations as a simulation ensemble. The following deterministic simulation and simulation ensembles are conducted based on the original HK-DEM and the aforementioned computational scenarios (table 6.1). They are named after the corresponding DEM and DEM realizations.

1. Deterministic simulation HK-DEM. One landslide process simulation run is conducted based on the original HK-DEM. This one time simulation corresponds to what is traditionally performed in a simulation-based hazard assessment study. The results serve as the basis to assess the impact of DEM uncertainty.

2. $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble. A total of 500 landslide process simulations are conducted based on the $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ DEM realizations. Each of them is referred to as $\text{USS}^n_{N=500}\{\text{RMSE} = 3.3, d = 180\}$, with $n = 1, 2, \ldots, 500$. This ensemble allows us to assess the impact of DEM uncertainty if only the RMSE is available.

3. $\text{CSS}_{N=500}$ ensemble. A total of 500 landslide process simulations are conducted based on the $\text{CSS}_{N=500}$ DEM realizations. Similar to (2), each of them is referred to as $\text{CSS}^n_{N=500}$. This ensemble allows us to assess the impact of DEM uncertainty if higher-accuracy reference data are available.

4. $\text{USS}_{N=500}\{\text{RMSE} = 0.5, 1.5, 2.5, d = 180\}$ ensembles. For each of the three different RMSE values, 500 landslide process simulations are conducted, while keeping the maximum autocorrelation length d constant. They lead to 1500 process simulations. The results allow us to discuss the unrepresentative RMSE issue as detailed in section 6.2.1. They can be also used to discuss the relationship between the degree of DEM uncertainty and its impact.

5. $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 0, 90, 270\}$ ensembles. For each of the three maximum autocorrelation length values, 500 landslide process simulations are conducted, while keeping the RMSE constant. They lead to 1500 process simulations. The results allow us to discuss the subjective d issue as detailed in section 6.2.1.

All in all this adds up to one deterministic simulation run HK-DEM, two simulation ensembles $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ and $\text{CSS}_{N=500}$ which are constructed from higher-accuracy reference data based on the 2m-DEM, and six additional ensembles $\text{USS}_{N=500}\{\text{RMSE} = 0.5, 1.5, 2.5, d = 180\}$ and $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 0, 90, 270\}$ to test sensitivities. Each ensemble consists of 500 process simulations. Each process simulation takes around 1 minute on a laptop with Intel Core i7-9750H CPU, adding up to around 67 hours of simulation time.

6.5 Results and discussions

This section is organized according to the simulation ensembles introduced in section 6.4.3. Section 6.5.1 presents the results of the deterministic simulation HK-DEM which serves as the basis for all following discussions. Section 6.5.2 is devoted to analyzing the impact of DEM uncertainty on landslide run-out simulation in the cases of only the RMSE being available ($\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble) and higher-accuracy reference data being available ($\text{CSS}_{N=500}$ ensemble). In section 6.5.3, the unrepresentative RMSE and subjective d issues are discussed based on the ensembles described in section 6.4.3 (labeled 4 and 5).

6.5.1 Deterministic simulation

Figures 6.7a-b show maximum flow height and velocity as given by the deterministic simulation HK-DEM. It should be noted that there is a relatively high elevation area at the end part of the channel in the HK-DEM as denoted within the red circle in figure 6.7b. It corresponds to the construction of debris-resisting barriers after the 2008 Yu Tung Road landslide as introduced in section 6.4.1. The flow material is decelerated and held back here. This point will be discussed later in section 6.5.2.1.

Besides maximum flow height and velocity, the apparent friction angle which is often used to characterize landslide run-out distance is discussed. It is defined as the angle of the line connecting the center of release mass and the center of deposit mass (De Blasio, 2011). The apparent friction angle evaluated from the deterministic simulation is 16.80°. This result is used as a reference to assess the impact of DEM uncertainty in the following discussions.

6.5.2 Unconditional and conditional stochastic simulations

While it is straightforward to present the results of a deterministic simulation run, a stochastic simulation-based ensemble of N simulation runs calls for tailored statistics to manage and interpret the extensive output data. First, the hazard probability P_l at a location is defined as the frequency of maximum flow height at that location exceeding a certain predefined height threshold value; hence

$$P_l = \frac{1}{N}\sum_{n=1}^{N} P_l^n,\ P_l^n = \begin{cases} 1, & \text{if } h_l^{\max,n} \geq \text{threshold} \\ 0, & \text{otherwise,} \end{cases} \tag{6.12}$$

where $h_l^{\max,n}$ denotes the maximum flow height at location l for the n-th simulation run of the corresponding ensemble. Hence P_l^n indicates whether location l is within the hazard area of the n-th simulation run for a given threshold, and P_l indicates the resulting hazard probability at location l considering the complete ensemble. Here, the threshold is set as 0.1 m, which matches the cutoff threshold of the deterministic simulation HK-DEM in figure 6.7a. Evaluation of hazard probabilities at all locations then gives rise to a probabilistic hazard map (Stefanescu et al., 2012), which provides an overall view of the DEM uncertainty impact.

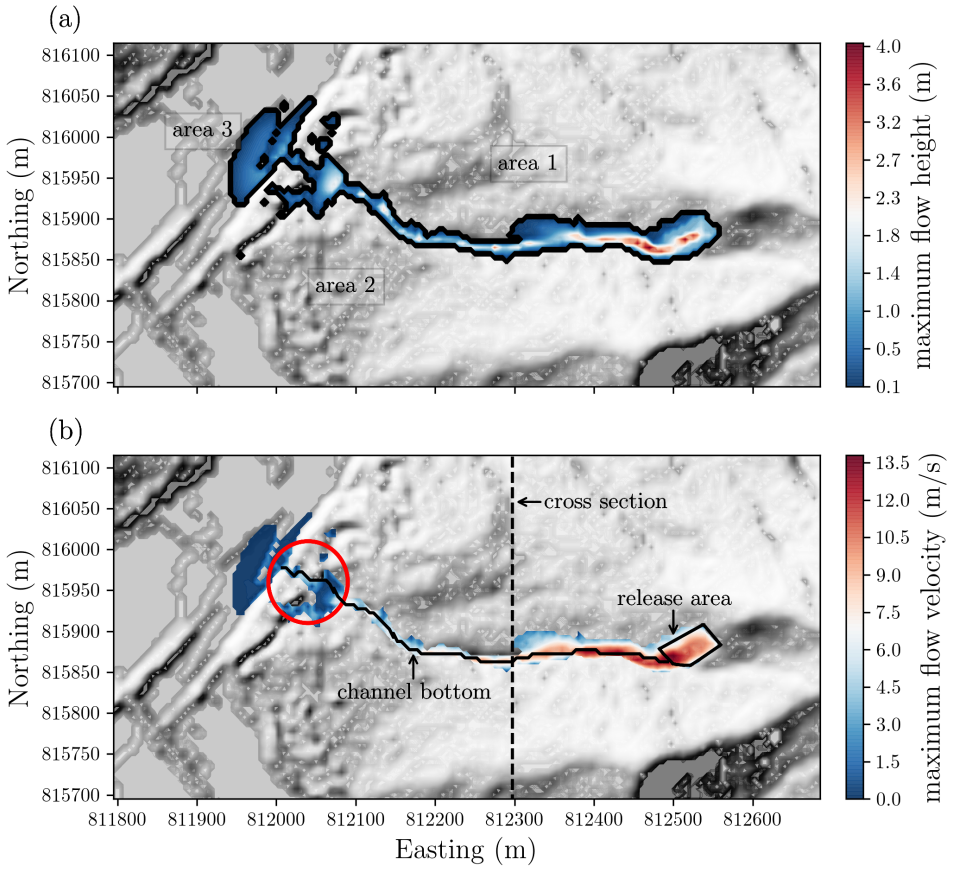

FIGURE 6.7: Results of the deterministic simulation HK-DEM. (a) Maximum flow height above a cutoff threshold of 0.1 m. The black outline is the 0.1 m contour of maximum flow height. The area within this outline is regarded as the hazard area. Areas 1-3 are denoted for later discussions (see section 6.5.2.1). (b) Maximum flow velocity above a cutoff threshold of 0.01 m/s. The relatively high elevation area within the red circle decelerates and holds back the flow material. The channel bottom and cross section are denoted for later discussions (see section 6.5.2.2).

Besides assessing the overall impact of DEM uncertainty in terms of the probabilistic hazard map, the local impact of DEM uncertainty on dynamic flow properties is also discussed, focusing on maximum flow height and velocity at locations along the channel bottom and the channel cross section denoted in figure 6.7b.

6.5.2.1 Probabilistic hazard maps

Figures 6.8a and 6.8c show the probabilistic hazard map for both the $\text{USS}_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble and the $\text{CSS}_{N=500}$ ensemble. It can be seen that the potential hazard area is much larger than the deterministic hazard area for both ensembles. The difference between the deterministic and the ensemble-based hazard area is most pronounced in areas 1-3 for the $\text{USS}_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble and in area 3 for the $\text{CSS}_{N=500}$ ensemble. Figures 6.8b and 6.8d show boxplots of the apparent friction angle distribution for both ensembles. It is evident that the apparent friction angle of both ensembles varies largely with respect to the apparent friction

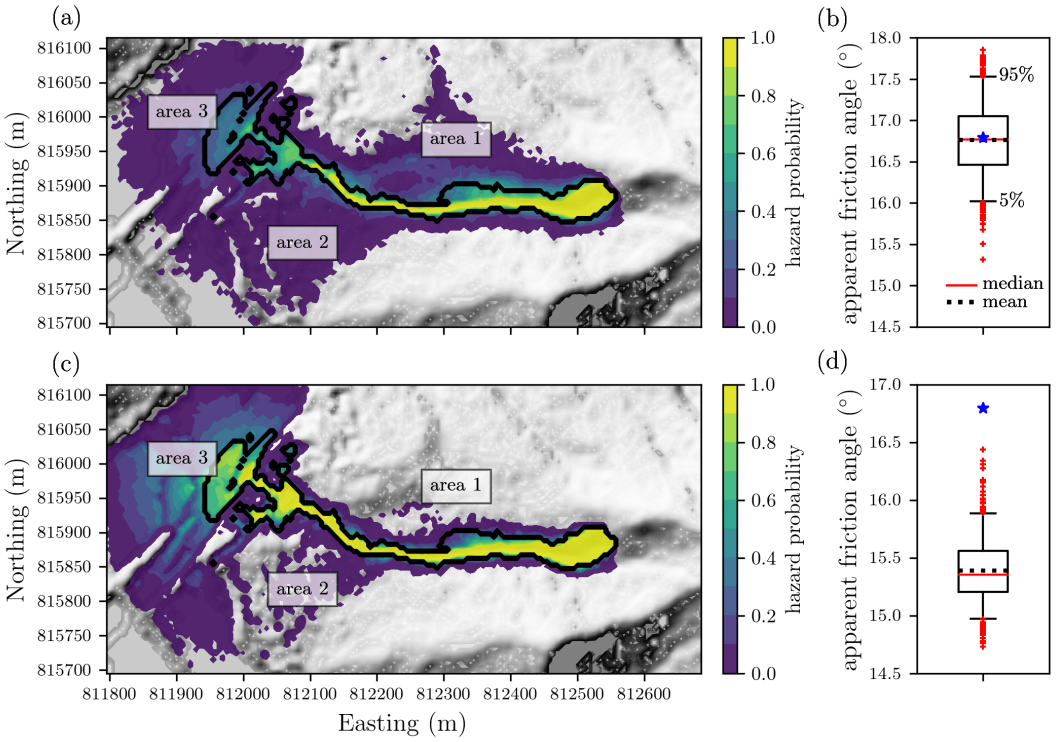

FIGURE 6.8: (a) Probabilistic hazard map and (b) corresponding apparent friction angle distribution of the $USS_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble; (c) probabilistic hazard map and (d) corresponding apparent friction angle distribution of the $CSS_{N=500}$ ensemble. The black outline plotted on the hazard maps represents the deterministic hazard area (see figure 6.7a). In the boxplots, the blue star denotes the apparent friction angle of the deterministic simulation HK-DEM. The difference between the deterministic and the ensemble-based hazard area is most pronounced in area 1-3 for the $USS_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble and in area 3 for the $CSS_{N=500}$ ensemble.

angle of the deterministic simulation (16.80°). The $CSS_{N=500}$ ensemble-based apparent friction angle (mean 15.39°) is smaller than the $USS_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble-based apparent friction angle (mean 16.76°).

As stated in section 6.3, analyzing terrain characteristics of the original DEM and DEM realizations may contribute to the interpretation of simulation results. By a preliminary analysis, no obvious relationships between landslide run-out simulation results and terrain characteristics were found at a specific location (on the cell level). One obvious reason for this is that a simulation result at one location is affected not only by terrain characteristics at the specific location but also by the complete upstream and surrounding terrain. Instead of discussing the effects of terrain characteristics at the cell level, the focus here is put on several compound terrain characteristics that contribute to understanding how DEM uncertainty may affect simulation results. The compound terrain characteristics include banks of the channel, especially the north bank near area 1 and south bank near area 2; the relatively high elevation area at the end part of the channel that holds back flow material as shown in figure 6.7b; topographic roughness; and the relatively flat area of area 3 (namely area with a relatively small slope).

Due to DEM uncertainty, topographic characteristics represented in DEM realizations vary from those represented in the original DEM. Specifically, firstly, topographic details of the deterministic channel tend to be dampened out in DEM realizations. The topographic details include banks of the channel as well as the relatively high elevation area at the end part of the channel that holds back flow material. Secondly, topographic roughness tends to increase.

Whether, where, and to what extent the topographic characteristics in DEM realizations would differ from the original DEM depend on (1) variability in DEM error – intuitively, the larger the variability, the more likely that topographic details of the deterministic channel would be dampened out and the larger the topographic roughness in DEM realizations – and (2) topographic details of the original DEM. If subject to the same DEM error, less well defined topographic characteristics in the original DEM are more likely to be changed in DEM realizations. For example, along the channel of the HK-DEM, the north bank of the channel near area 1 and the south bank of the channel near area 2 are less well defined compared to other parts of the channel banks. Flow material could be more easily diverted to area 1 and area 2 where elevations are relatively low and some local small channels exist. Area 3 could also be regarded as less well defined since it is relatively flat and thus is sensitive to DEM uncertainty (Temme et al., 2009).

The change in each topographic characteristic has a corresponding impact on landslide run-out behavior. Specifically (1) if banks of the deterministic channel are dampened out in DEM realizations, flow material tends to spread out along the channel cross-section direction and travel distance is shorter; (2) if the relatively high elevation area that holds back flow material is dampened out, flow material tends to travel further; and (3) increasing topographic roughness leads to higher simulated momentum losses and shorter travel distance as pointed out by McDougall (2017).

For the $\mathrm{USS}_{N=500}$ $\{\mathrm{RMSE} = 3.3, d = 180\}$ ensemble, the variability in DEM error is relatively large, i.e., around 3.3 m governed by the non-bias-corrected RMSE based on $\varepsilon_k^\star\{k = 180\}$ (see figure 6.6c). In this situation, both the north bank near area 1 and south bank near area 2 as well as the relatively high elevation area at the end part of the channel can be dampened out in HK-DEM realizations. For the $\mathrm{CSS}_{N=500}$ ensemble, the variability in DEM error is relatively small, i.e., around 1.5 m governed by the standard deviation (σ_ε) based on $\varepsilon_k^\star\{k = 180\}$ (see figure 6.6d). In this situation, the banks tend to remain well defined, while the relatively high elevation area can be dampened out in HK-DEM realizations. Thus, area 1 and area 2 are possibly subject to hazard in the $\mathrm{USS}_{N=500}$ $\{\mathrm{RMSE} = 3.3, d = 180\}$ ensemble but less likely to be so in the $\mathrm{CSS}_{N=500}$ ensemble. As mentioned above, area 3 is a flat area which is sensitive to DEM uncertainty. Furthermore, it is located near the deposition, around which the impact of upstream DEM uncertainty seems to accumulate. Thus, it is highly affected in both ensembles.

The apparent friction angle distribution is determined by a combined effect of change in channel banks, change in the relatively high elevation area at the end part

of the channel, and increasing topographic roughness. For the $\text{USS}_{N=500}$ {RMSE $=$ $3.3, d = 180$} ensemble, a deteriorated channel bank representation and increasing topographic roughness make flow material travel a shorter distance, i.e., larger apparent friction angle, while a deteriorated relatively high elevation area representation allows flow material to travel further, i.e., smaller apparent friction angle. For the $\text{CSS}_{N=500}$ ensemble, channel banks are likely to remain well defined and the degree of topographic roughness increase is lower due to its relatively small variability in DEM error compared to the $\text{USS}_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble. Thus, flow material in the $\text{CSS}_{N=500}$ ensemble tends to travel a longer distance, i.e., smaller apparent friction angle, compared to the flow material in the $\text{USS}_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble.

In summary, it can be concluded from the probabilistic hazard maps and boxplots of apparent friction angle distribution that (1) accounting for DEM uncertainty may significantly increase the potential hazard area, (2) the potential hazard area is highly related to the variability in DEM error and topographic characteristics of the original DEM, and (3) USS based on the RMSE only may overestimate the spread of potential hazard area and underestimate travel distance due to a non-bias-corrected RMSE that overestimates the variability in DEM error.

It should be noted that the probabilistic hazard map here is constructed based on maximum height and a predefined threshold. In simulation-based hazard assessment practice, one may indicate potential hazard using other indicators, e.g., the maximum momentum that reflects the impact pressure, and correspondingly modify the threshold value. In this case, the workflow is easily extendible to account for other indicators.

6.5.2.2 Dynamic flow properties

Figures 6.9a, 6.9c, and 6.9e show elevation, maximum flow height, and maximum flow velocity at locations along the channel bottom based on the $\text{USS}_{N=500}$ {RMSE $=$ $3.3, d = 180$} ensemble. It is evident that both maximum height and maximum velocity at these locations largely vary from those of the deterministic simulation. Specifically, the mean of maximum height (maximum velocity) values at all the locations based on the deterministic simulation is 1.28 m (7.17 m/s). The mean of the ensemble-based 90 % credible interval of maximum height (maximum velocity) is [0.18 m, 2.17 m] ([0.99 m/s, 7.89 m/s]; the range between the mean of the ensemble-based 5th percentile and the mean of the ensemble-based 95th percentile). Another observation is that the ensemble-based mean of flow dynamic properties is generally smaller than the mean of flow dynamic properties of the deterministic simulation (as seen by the dashed red line being generally under the black line in both figures 6.9c and 6.9e). The mean of the ensemble-based mean of maximum height (maximum velocity) is 0.85 m (4.57 m/s), around 66% (64%) of the mean of the deterministic simulation at 1.28 m (7.17 m/s; see figures 6.9c and 6.9e).

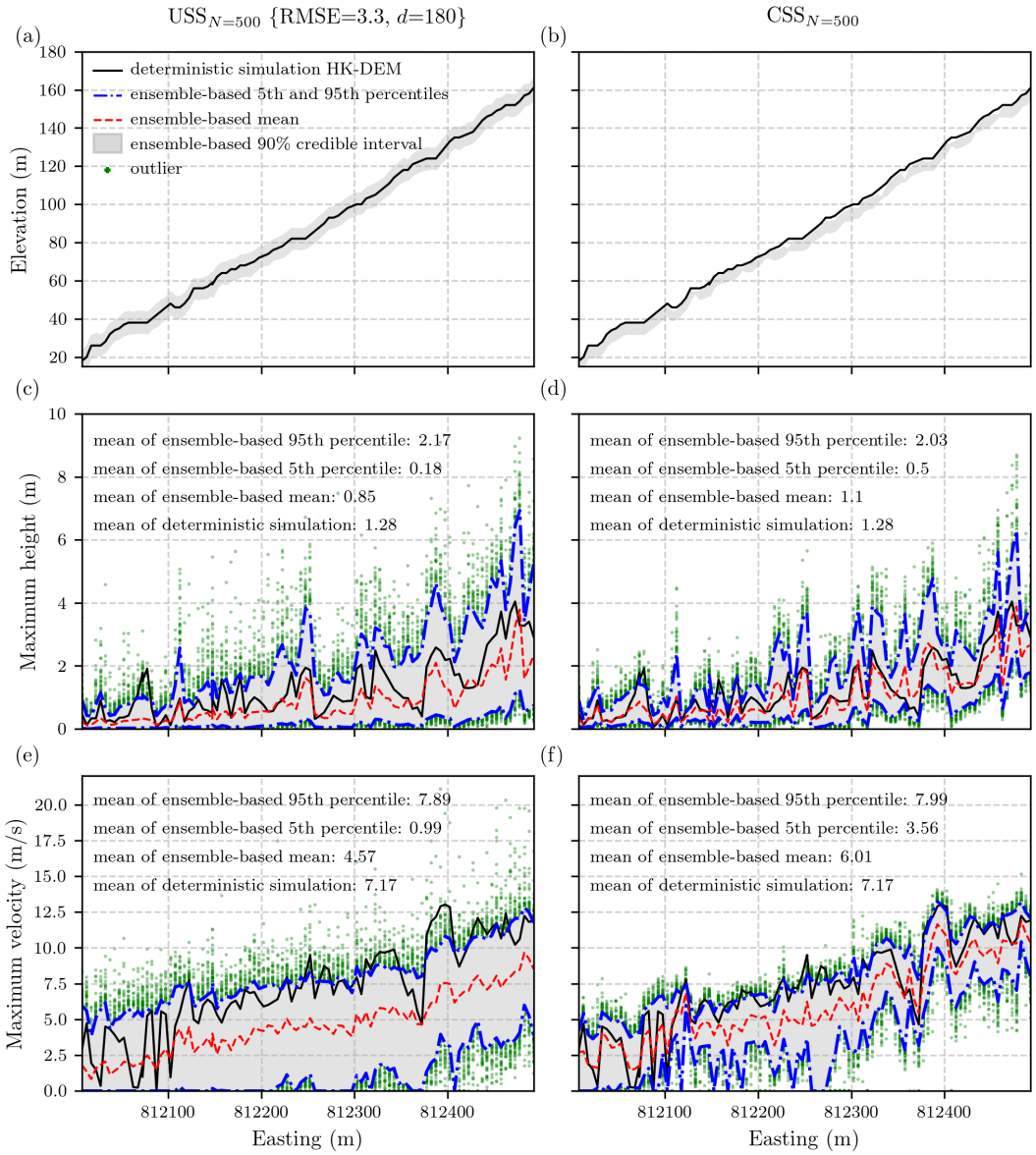

FIGURE 6.9: Elevation, maximum height, and maximum velocity at locations along the channel bottom (see figure 6.7b). Panels (a), (c), and (e) correspond to the USS$_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble and panels (b), (d), and (f) to the CSS$_{N=500}$ ensemble. In each panel, dash-dotted blue lines represent ensemble-based 5th and 95th percentiles of the quantity. The dashed red line represents the ensemble-based mean of the quantity. The black line denotes corresponding results of the deterministic simulation. Annotated mean values are an average of all the locations. Ensemble-based flow dynamic properties largely vary from deterministic simulation results. The variation range of the USS$_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble is larger, while its ensemble-based mean is smaller, compared to counterparts of the CSS$_{N=500}$ ensemble.

Figures 6.9b, 6.9d, and 6.9f show corresponding results based on the CSS$_{N=500}$ ensemble. Similar trends to those in the USS$_{N=500}${RMSE $= 3.3, d = 180$} ensemble can also be observed. Namely, both maximum height and maximum velocity at these locations largely vary from those of the deterministic simulation; the ensemble-based mean of flow dynamic properties is generally smaller than that of the deterministic results. The main differences are that the variation range of CSS$_{N=500}$

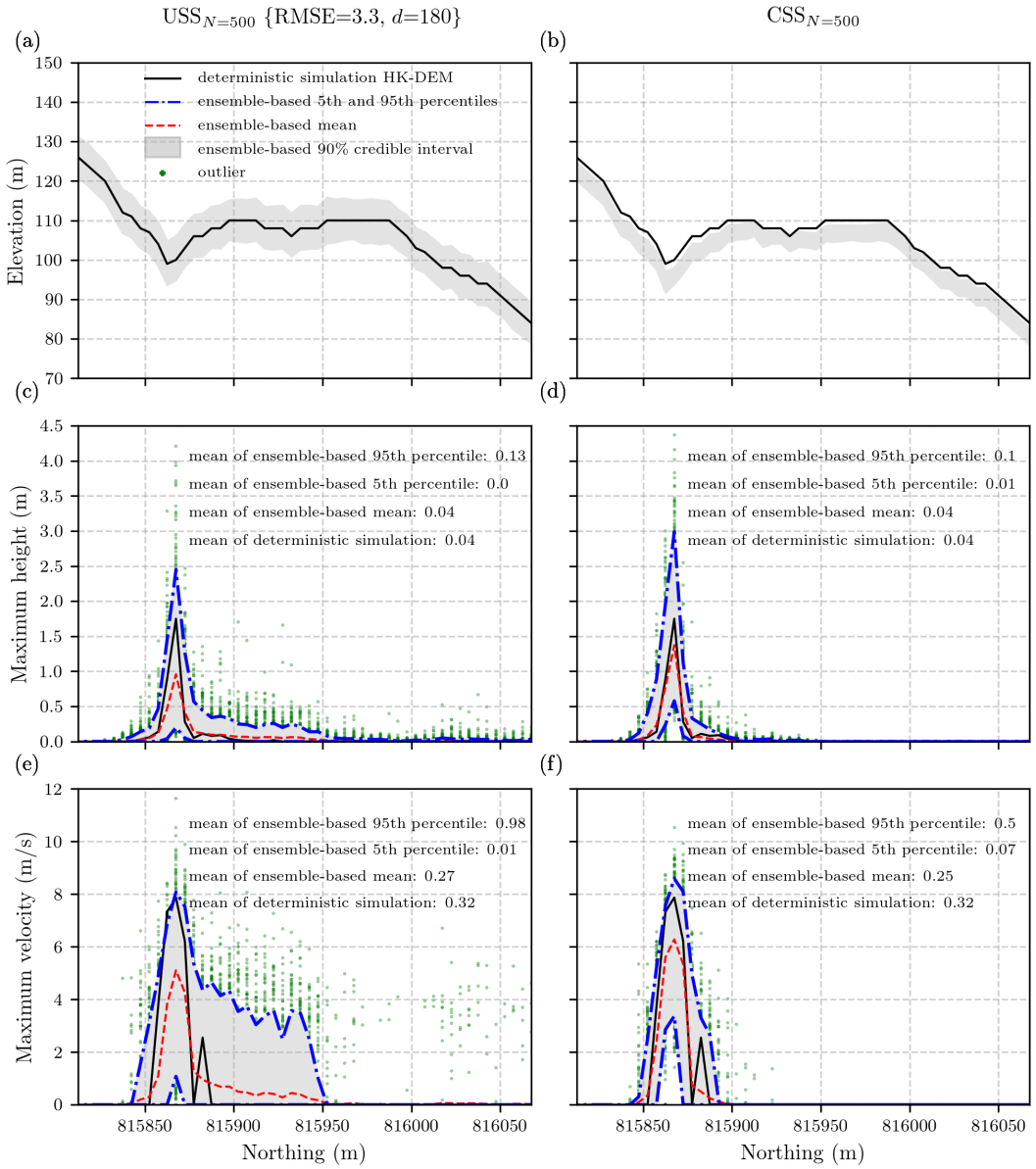

FIGURE 6.10: Elevation, maximum height, and maximum velocity at locations along the channel cross section (see figure 6.7b). Panels (a), (c), and (e) correspond to the USS$_{N=500}${RMSE $= 3.3, d = 180$} ensemble and panels (b), (d), and (f) to the CSS$_{N=500}$ ensemble. Due to DEM uncertainty, flow material of both ensembles tends to spread out along the channel cross-section direction. The ensemble-based mean of flow dynamic properties at the channel bottom location is smaller than flow dynamic properties at the channel bottom location of the deterministic simulation (compare peak value of dashed red line with peak value of black line). The more the flow material spreads out, the smaller the ensemble-based mean of flow dynamic properties at the channel bottom location (compare results of the USS$_{N=500}${RMSE $= 3.3, d = 180$} ensemble with those of the CSS$_{N=500}$ ensemble).

ensemble-based flow dynamic properties is smaller and the CSS$_{N=500}$ ensemble-based mean of flow dynamic properties is larger compared to those of the USS$_{N=500}$ {RMSE $= 3.3, d = 180$} ensemble. More specifically, the mean of the CSS$_{N=500}$ ensemble-based 90% confidence interval of maximum height (maximum velocity) is [0.5 m, 2.03 m] ([3.56 m/s, 7.99 m/s]). The mean of the CSS$_{N=500}$ ensemble-based

mean of maximum height (maximum velocity) is 1.1 m (6.01 m/s), around 86% (84%) of the mean of the deterministic simulation 1.28 m (7.17 m/s; see figures 6.9d and 6.9f).

The above observations result from similar factors to those discussed in section 6.5.2.1. Due to DEM uncertainty, the following statements can be made:

- Ensemble-based flow dynamic properties are likely to vary from those of the deterministic simulation. Larger variability in DEM error is likely to result in more extreme results. As discussed in section 6.5.2.1, the variability in DEM error for the $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble is larger than that for the $\text{CSS}_{N=500}$ ensemble due to the unrepresentative RMSE issue. Thus the variation range of $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble-based flow dynamic properties is generally larger than that of $\text{CSS}_{N=500}$ ensemble-based flow dynamic properties, giving a larger mean of the ensemble-based 90% credible interval (the trend would be more clear if we also consider outliers outside the 90% credible interval).

- Banks of the deterministic channel may be dampened out in DEM realizations. Deteriorated channel bank representation makes flow material more spread out along the channel cross-section direction. This could lead to a smaller ensemble-based mean of flow dynamic properties at channel bottom locations compared to flow dynamic properties of the deterministic simulation. This can be directly seen in figure 6.10, which displays results of one channel cross section. Also, due to larger variability in DEM error, flow material in the $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble is more spread along the channel cross-section direction, resulting in a smaller ensemble-based mean of flow dynamic properties at channel bottom locations compared to that of the $\text{CSS}_{N=500}$ ensemble. This can also be seen in figure 6.10.

- Topographic roughness in DEM realizations tends to increase. As pointed out in section 6.5.2.1, increasing topographic roughness results in higher simulated momentum losses and thus smaller flow dynamic properties on average. The higher the degree of topographic roughness increase, the higher the simulated momentum losses and the smaller the flow dynamic properties. This also contributes to a smaller ensemble-based mean of flow dynamic properties at channel bottom locations compared to flow dynamic properties of the deterministic simulation, as well as to a smaller $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$ ensemble-based mean of flow dynamic properties at channel bottom locations compared to the $\text{CSS}_{N=500}$ ensemble.

Based on the ensembles' dynamic flow properties, it can be concluded that (1) accounting for DEM uncertainty may significantly affect dynamic flow properties, e.g., maximum height and maximum velocity, and hence any hazard assessment that is based on landslide dynamics and (2) USS based only on the RMSE may overestimate the range of dynamic flow properties and underestimate the ensemble-based

mean of dynamic flow properties due to an unrepresentative RMSE that overestimates the variability in DEM error.

6.5.3 Impact of RMSE and correlation length on USS

The unrepresentative RMSE and subjective d issues as introduced in section 6.2.1 are discussed here based the ensembles $\text{USS}_{N=500}\{\text{RMSE} = 0.5, 1.5, 2.5, d = 180\}$, $\{\text{RMSE} = 3.3, d = 0, 90, 270\}$, and $\text{USS}_{N=500}\{\text{RMSE} = 3.3, d = 180\}$. Results of the $\text{CSS}_{N=500}$ ensemble are used as a reference since $\text{CSS}_{N=500}$ incorporated more information on the DEM error. It is thus reasonable to assume that its results reflect the reality better.

Figure 6.11 shows the consolidated results of the ensembles. Focusing on the left column, it can be seen that with increasing RMSE (1) low-probability (0-0.2) hazard area significantly increases and high-probability (0.8-1) hazard area gradually decreases, leading to increase in overall potential hazard area (using the same threshold value); (2) except for the RMSE $= 0.5$ m ensemble, the apparent friction angle steadily increases; and (3) the range of extreme values of maximum height (maximum velocity) at channel bottom locations increases while the average of maximum height (maximum velocity) at channel bottom locations decreases.

For purely RMSE-based USS, the standard deviation of DEM error is assumed to be determined by the RMSE. Hence larger RMSE indicates larger variability in DEM error in DEM realizations. The larger the variability in DEM error, the more likely topographic details of the deterministic channel would be dampened out and the larger the topographic roughness in DEM realizations. As discussed in section 6.5.2.1, this would make flow material more spread out along the channel cross-section direction (namely larger potential hazard area) and travel a shorter distance (namely larger apparent friction angle). As discussed in section 6.5.2.2, larger variability in DEM error is likely to result in more extreme values of flow dynamic properties (namely larger range of extreme values), while spreading of flow material along the channel cross-section direction and larger topographic roughness lead to a smaller ensemble-based mean of flow dynamic properties at channel bottom locations.

As discussed in section 6.5.2.1, the apparent friction angle distribution is determined by a combined effect of change in channel banks, change in the relatively high elevation area at the end part of the channel, and increasing topographic roughness. It naturally follows that for a small variability in DEM error (here RMSE $= 0.5$ m), all the changes are less significant in DEM realizations, and thus the apparent friction angle of the $\text{USS}_{N=500}\{\text{RMSE} = 0.5, d = 180\}$ ensemble closely matches the deterministic simulation result. For an intermediate variability in DEM error (here RMSE $= 1.5$ m), the relatively high elevation area at the end part of the channel is subject to change, while channel banks tend to remain well defined in DEM realizations. This leads to a longer travel distance of the $\text{USS}_{N=500}\{\text{RMSE} = 1.5, d = 180\}$

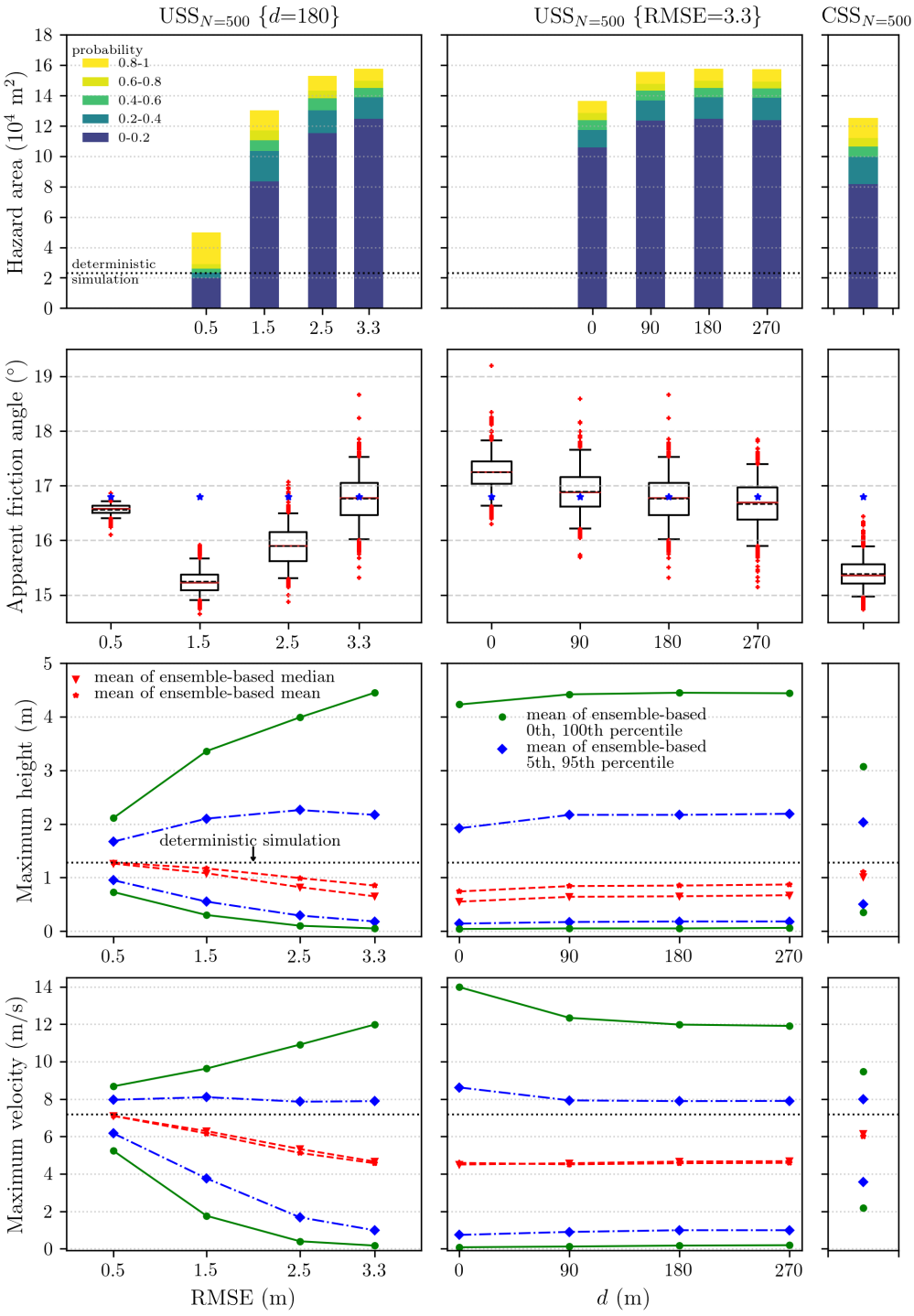

FIGURE 6.11: Consolidated results of all ensembles. The left, middle, and right columns correspond to ensembles USS$_{N=500}${RMSE $= 0.5, 1.5, 2.5, 3.3$, $d = 180$}, ensembles USS$_{N=500}${RMSE $= 3.3, d = 0, 90, 180, 270$}, and the ensemble CSS$_{N=500}$, respectively. The first row shows stacked bar plots of the potential hazard area's magnitude based on the probabilistic hazard map for each ensemble (see figures 6.8a and 6.8c). The second row shows apparent friction angle distribution. The last two rows show statistics of maximum height and maximum velocity at channel bottom locations (see figures 6.9c-f).

ensemble (namely smaller apparent friction angle) in comparison to that of the deterministic simulation result.

From the middle column of figure 6.11, the results for a USS ensemble of vanishing spatial autocorrelation $\mathrm{USS}_{N=500}\{\mathrm{RMSE} = 3.3, d = 0\}$ consistently differ significantly from USS ensembles that include spatial autocorrelation, namely $\mathrm{USS}_{N=500}\{\mathrm{RMSE} = 3.3, d = 90, 180, 270\}$ ensembles. This indicates that whether spatial autocorrelation is considered or not may make a difference but the extent of spatial autocorrelation has less influence on simulation results. Since spatial autocorrelation exists in topographic data but there is often a lack of information on its exact autocorrelation length, this is actually good news for practical hazard assessment studies.

Comparing the left column of figure 6.11 with the right column, it can furthermore be seen that the results of the $\mathrm{USS}_{N=500}\{\mathrm{RMSE} = 1.5, d = 180\}$ ensemble are quite close to the results of the $\mathrm{CSS}_{N=500}$ ensemble. The $\mathrm{USS}_{N=500}\{\mathrm{RMSE} = 1.5, d = 180\}$ ensemble is informed of the bias-corrected RMSE (namely the true standard deviation, in our case 1.5 m; see figure 6.3b). It indicates that if a bias-corrected RMSE is given, USS is able to provide reasonable results considering that the extent of spatial autocorrelation has less influence on simulation results.

All in all, it can be found that (1) the results of USS are in general more sensitive to values of the RMSE and less sensitive to values of d; (2) an unrepresentative RMSE that overestimates the variability in DEM error may overestimate the potential hazard area and extreme values of dynamic flow properties; (3) whether or not spatial autocorrelation of DEM error is considered can make a difference in ensemble-based simulation results; and (4) if a bias-corrected RMSE is given, it is possible to obtain reasonable ensemble-based simulation results using USS.

6.6 Dimensionality reduction of topographic uncertainty

In both geostatistical methods (USS and CSS) introduced in section 6.2, the DEM error field $\mathcal{E}_{mn}$ is implicitly or explicitly modeled by Gaussian processes with two-dimensional input $\mathbf{s} = (s_x, s_y) \in \mathbb{R}^2$, namely

$$g(\cdot) \sim \mathcal{GP}(m(\cdot), C(\cdot, \cdot)), \tag{6.13}$$

where $g(\cdot)$ represents the unknown function that maps the two-dimensional spatial input $\mathbf{s}$ to the scalar DEM error ε at $\mathbf{s}$. Gaussian processes with two-dimensional input are also known as Gaussian random fields (Bishop, 2006). In USS, due to the lack of higher-accuracy reference data, the mean $m(\cdot)$ is simply assumed to be zero; the variance in the covariance function $C(\cdot, \cdot)$ is assumed to be given as the square of RMSE and the correlation length parameter in the covariance function is subjectively chosen. In CSS, the Gaussian process is updated to a Gaussian process posterior based on higher-accuracy reference data. The hyperparameters can be learned using the maximum likelihood estimation as presented in section 3.3.1.

Sampling realizations of the DEM error field $\mathcal{E}_{mn}$ in both USS and CSS involves

$m \times n$ random variables. The sampling itself can be managed using the spatial moving averages technique (section 6.2.1) or the sequential sampling algorithm (section 6.2.2). The high dimensionality ($m \times n$) however hinders us from building GP emulators (emulating relations between high-dimensional DEM uncertainty and landslide run-out simulation outputs) in following uncertainty quantification analyses, since GP emulation is not suitable for high-dimensional input functions. This may cause two issues: first, computational cost of DEM uncertainty quantification purely based on simulation runs can be expensive, especially when a single simulation run takes relatively long time; second, it makes simultaneous quantification of high-dimensional DEM uncertainty and other low-dimensional uncertain factors (like initial distribution of the flow mass, rheological parameters) difficult in the unified framework. Here, I propose a new method that makes GP emulation applicable also for DEM uncertainty by dimensionality reduction.

Given the Gaussian process modeling the DEM error field $\mathcal{E}_{mn}$ (equation 6.13), any sample (realization) from it can be expressed using the Karhunen–Loève expansion as follows:

$$g(\mathbf{s}) = m(\mathbf{s}) + \sum_{i=1}^{\infty} w_i \sqrt{\lambda_i} \eta_i(\mathbf{s}), \tag{6.14}$$

where $w_i \sim \mathcal{N}(0,1), i = 1, \ldots, \infty$, are independent standard Gaussian random variables; λ_i and $\eta_i(\mathbf{s})$ are eigenvalues and eigenfunctions of the covariance function obtained from the Fredholm integral equation (Uribe et al., 2020)

$$\int C(\mathbf{s}, \mathbf{s}') \eta_i(\mathbf{s}') d\mathbf{s}' = \lambda_i \eta_i(\mathbf{s}). \tag{6.15}$$

We can easily prove that equation 6.14 gives samples from the Gaussian process by checking the mean and the covariance as follows (Liu et al., 2019):

$$E[g(\mathbf{s})] = m(\mathbf{s}) + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \eta_i(\mathbf{s}) E[w_i] = m(\mathbf{s}), \tag{6.16}$$

$$\begin{aligned} E[(g(\mathbf{s}) - m(\mathbf{s}))(g(\mathbf{s}') - m(\mathbf{s}'))] &= E[\sum_{i=1}^{\infty} w_i \sqrt{\lambda_i} \eta_i(\mathbf{s}) \sum_{j=1}^{\infty} w_j \sqrt{\lambda_j} \eta_j(\mathbf{s}')] \\ &= \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \sqrt{\lambda_i \lambda_j} \eta_i(\mathbf{s}) \eta_j(\mathbf{s}') E[w_i w_j] \\ &= \sum_{i=1}^{\infty} \lambda_i \eta_i(\mathbf{s}) \eta_i(\mathbf{s}') = C(\mathbf{s}, \mathbf{s}'). \end{aligned} \tag{6.17}$$

The last step in equation 6.17 is based on the Mercer's theorem (Uribe et al., 2020).

In order to reduce the dimensionality, the Karhunen–Loève expansion (equation 6.14) needs to be truncated after a limited number (t) of terms, namely

$$g(\mathbf{s}) \approx m(\mathbf{s}) + \sum_{i=1}^{t} w_i \sqrt{\lambda_i} \eta_i(\mathbf{s}). \tag{6.18}$$

The number t can be determined for example by investigating the decay of eigenvalues (Liu et al., 2019). After the truncation, generating realizations of the DEM error field $\mathcal{E}_{mn}$ reduces to sample the t independent standard Gaussian random variables $w_i, i = 1, \ldots, t$. It means that the high-dimensional ($m \times n$) DEM uncertainty can

Algorithm 5 Emulator-based uncertainty quantification

1: Construct a Gaussian process to model the DEM error field $\mathcal{E}_{mn}$.
2: Apply the Karhunen–Loève expansion to the Gaussian process.
3: Reduce the dimensionality of DEM uncertainty to a limited number of random variables $w_i, i = 1, \ldots, t$, by truncating the Karhunen–Loève expansion series.
4: Choose training input points $\mathbf{x}^{tr} = \{\mathbf{x}_i\}_{i=1}^{n^{tr}}$ using a Latin hypercube design. Each input point represents a configuration of $w_i, i = 1, \ldots, t$, and other concerned low-dimensional uncertain inputs like the release volume v_0, rheological parameters μ, ξ, etc.
5: Run the simulator $f(\mathbf{x})$, namely the landslide run-out model, at each chosen input point to obtain $\mathbf{y}^{tr}$.
6: Build the emulator $\hat{f}(\mathbf{x})$ based on $\mathbf{x}^{tr}$-$\mathbf{y}^{tr}$.
7: Set the Monte Carlo sample size N and realization sample size N_r. Sample input points $\{\mathbf{x}_i\}_{i=1}^{N}$ from the input domain using Monte Carlo-type sampling schemes.
8: **for** $n_r = 1, \ldots, N_r$ **do**
9: Sample a realization at input points $\{\mathbf{x}_i\}_{i=1}^{N}$ from the emulator $\hat{f}(\mathbf{x})$, denoted as $\{\tilde{f}^{n_r}(\mathbf{x}_i)\}_{i=1}^{N}$.
10: **end for**
11: **return** $\{\tilde{f}^{n_r}(\mathbf{x}_i)\}_{i=1,\ldots,N}^{n_r=1,\ldots,N_r}$.
12: Quantify uncertainty of the concerned model output y by statistically analyzing $\{\tilde{f}^{n_r}(\mathbf{x}_i)\}_{i=1,\ldots,N}^{n_r=1,\ldots,N_r}$, like calculating the mean, standard deviation, or probability distribution.

now be represented by a limited number of variables $w_i, i = 1, \ldots, t$.

Armed with the Karhunen–Loève expansion, DEM uncertainty can then be studied together with other low-dimensional uncertain factors in the unified emulator-based framework. More specifically, a Gaussian process that models the DEM error field $\mathcal{E}_{mn}$ is first constructed using USS or CSS depending on available information on the DEM error. Then the Karhunen–Loève expansion is applied, leading to a limited number of random variables $w_i, i = 1, \ldots, t$, representing the DEM uncertainty. In the next step, n^{tr} training input points are sampled from the input space consisting of $w_i, i = 1, \ldots, t$, and other low-dimensional uncertain factors (like μ, ξ, v_0, etc.). Corresponding n^{tr} landslide run-out simulations are conducted and training outputs are extracted. Based on the training data, GP emulators are built. Finally, uncertainty quantification of all the uncertain factors can be performed based on trained GP emulators. Algorithm 5 summarizes the proposed method and figure 6.12 provides a schematic illustration.

The same concept can also be applied to the emulator-based global sensitivity analysis method developed in chapter 4. Namely, by first reducing its dimensionality using the Karhunen–Loève expansion, DEM uncertainty can be treated the same as other low-dimensional uncertain factors in an emulator-based global sensitivity analysis. It should be noted that even the proposed Karhunen–Loève expansion-based method is attractive and promising, its implementation and application are yet planned for future work.

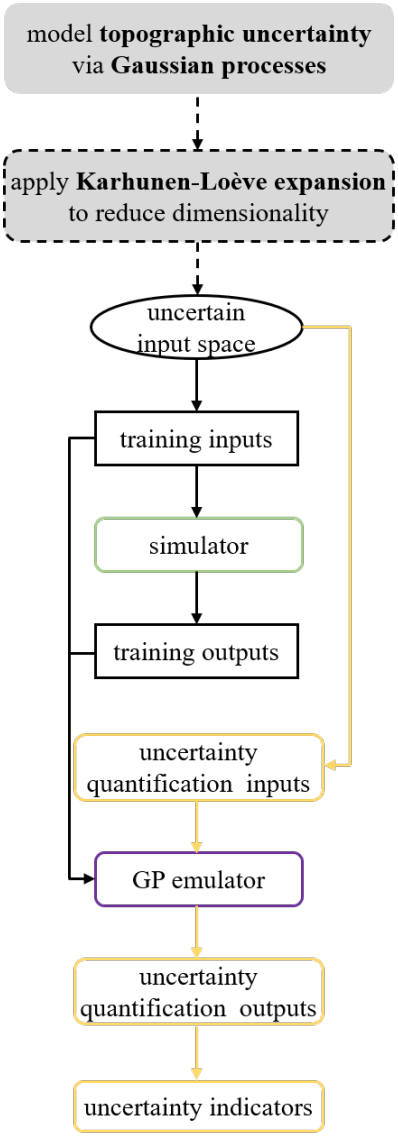

FIGURE 6.12: A schematic representation of the emulator-based Monte Carlo-type uncertainty quantification method (see figure 6.1) extended to high-dimensional topographic uncertainty. The dashed box and arrows indicate that the Karhunen–Loève expansion is proposed yet not implemented and applied in this thesis.

6.7 Summary

In this chapter, two types of geostatistical methods, namely unconditional and conditional stochastic simulations, have been introduced and employed to study the impact of topographic uncertainty on landslide run-out analyses. The case study is based upon a historic landslide event for which two DEM data sets of different accuracy are available. The computational workflow including the stochastic simulation to generate the DEM realizations, landslide run-out modeling, statistical analysis and visualization, and terrain analysis has been implemented in the unified framework in a modular way. How DEM uncertainty propagates in the results of landslide run-out analyses is discussed in detail. In addition, the two major issues of purely RMSE-based unconditional stochastic simulation (unrepresentative RMSE

and subjective d) have been addressed. Furthermore, a new method based on the Karhunen–Loève expansion is proposed in order to make GP emulation also applicable for high-dimensional DEM uncertainty. While being attractive and promising, its implementation and application are yet planned for future work. The main findings are as follows:

- DEM uncertainty significantly affects landslide run-out modeling depending on how well the underlying flow path is represented, which is determined by topographic characteristics of the original DEM and the variability in DEM error. For the same degree of variability in DEM error, the less well defined parts of the flow path in the original DEM are more likely to be affected and thus lead to change in flow behavior at these parts. Also, an increasing variability in DEM error leads to an increased impact. More specifically, with increasing variability in the DEM error, the potential hazard area and extreme values of dynamic flow properties are likely to increase. This shows the importance of considering topographic uncertainty for simulation-based landslide hazard assessment rather than simply trusting the results of a deterministic simulation if a high accuracy of the DEM source is not guaranteed. Also, a preliminary terrain analysis may give some hints on areas that will potentially be affected by a topographic uncertainty study.

- Both unconditional and conditional stochastic simulation methods can be applied to study DEM uncertainty propagation in landslide run-out modeling. Their main difference is that unconditional stochastic simulation can be conducted based on RMSE information only, while conditional stochastic simulation requires the availability of higher-accuracy reference data and is thus more accurate. The higher-accuracy reference data provide additional information on the DEM error structure, i.e., its spatial autocorrelation. If the DEM does not contain systematic bias or alternatively a bias-corrected RMSE is provided, the unconditional stochastic simulation yields reasonable results. Otherwise, the assumptions underlying the unconditional stochastic simulation lead to an overestimation of the DEM error variability, which leads to an increased potential impact of DEM uncertainty on the potential hazard area and to extreme values of dynamic flow properties. In particular, the study shows that if no higher-accuracy reference data are available or if computational costs of a conditional stochastic simulation are too large, the results of a RMSE-based unconditional stochastic simulation can still be used to provide an upper bound on the potential hazard area as well as extreme values of flow dynamic properties for a hazard assessment to take topographic uncertainties into account.

- Results of an unconditional stochastic simulation are in general sensitive to the RMSE value as well as sensitive to whether or not the DEM error's spatial autocorrelation is considered. If the latter is taken into account, results

are less sensitive to the actual value of the DEM error's maximum autocorrelation length. This indicates that determining a representative RMSE may be more important than finding a correct maximum autocorrelation length, while the DEM error's spatial autocorrelation should not be ignored for simulation-based landslide hazard assessment.

6.8 Modifications to the published paper

As mentioned in the outline (section 1.3), this chapter is a modified version of the published paper: Hu Zhao, Julia Kowalski. Topographic uncertainty quantification for flow-like landslide models via stochastic simulations, *Natural Hazards and Earth System Sciences*, 20, 1441-1461, 2020. Regarding the published paper, HZ and JK conceived the idea and designed the case study. HZ performed the implementation, simulations, and result analyses with contributions from JK. HZ wrote the manuscript. JK reviewed and revised the manuscript. Two anonymous reviewers reviewed the manuscript. Paola Reichenbach edited the manuscript.

 The main modifications to the published paper are listed below.

To avoid redundancies, the following parts of the published paper are left out in this chapter: (1) the descriptions of landslide hazards and landslide run-out models in the introduction of the published paper. They have been introduced in chapter 1 and chapter 2 of this thesis; (2) the mathematical details of landslide run-out models in section 2 of the published paper. They have been integrated with chapter 2 of this thesis.

To improve the coherence of this thesis as a whole, the descriptions of the following parts of the published paper are modified in this chapter: (1) the two geostatistical methods in section 3 of the published paper (corresponding to section 6.2); (2) the implementation in section 4 of the published paper (corresponding to section 6.3).

To include extended work after the publication, the following parts which are not included in the published paper are added in this chapter: (1) the introductory paragraphs before section 6.1. They link the published paper (focusing on topographic uncertainty quantification) to the overall uncertainty quantification of landslide run-out models; (2) the content of section 6.6.

Chapter 7

Concluding remarks and future work

7.1 Concluding remarks

This thesis presented a unified framework that allows us to systematically, routinely, and efficiently investigate various uncertainties associated with landslide run-out models, as shown in figure 7.1. The framework was built upon two essential components, namely physics-based landslide run-out models and data-driven Gaussian process emulators. Theories of physics-based landslide run-out models and Gaussian processes were introduced in detail in chapter 2 and chapter 3 respectively. Advanced statistical tools including the Sobol' sensitivity analysis, Bayesian active learning, and Monte Carlo-type uncertainty quantification, were integrated with the two core components. Within the unified framework, both forward and inverse problems resulting from the various uncertain factors can be comprehensively and efficiently studied. While the framework could be potentially applied to any landslide run-out model owing to its data-driven nature, its implementation and application in this thesis were focused on depth-averaged shallow flow type models with the Voellmy rheology. More specifically, the open-source run-out solver r.avaflow (running on LINUX operating systems) and the commercial run-out solver RAMMS (running on WINDOWS operating systems) were integrated in the unified framework and used for the case studies.

Chapter 4 presented a new methodology that enables efficient variance-based global sensitivity analyses of computationally expensive landslide run-out models. The methodology was developed by coupling depth-averaged shallow flow type run-out models, Sobol' sensitivity analysis, robust multivariate Gaussian process emulation techniques, and an algorithm accounting for the emulator-induced uncertainty. A Python-based workflow was implemented in the unified framework to realize the methodology. It allows us to efficiently leverage existing open-source software and packages including the landslide run-out solver r.avaflow, robust Gaussian process emulation package RobustGaSP, and the sensitivity analysis package SALib. The proposed method was illustrated and validated by a case study based on the 2017 Bondo landslide event. In addition, the sensitivity of selected run-out model

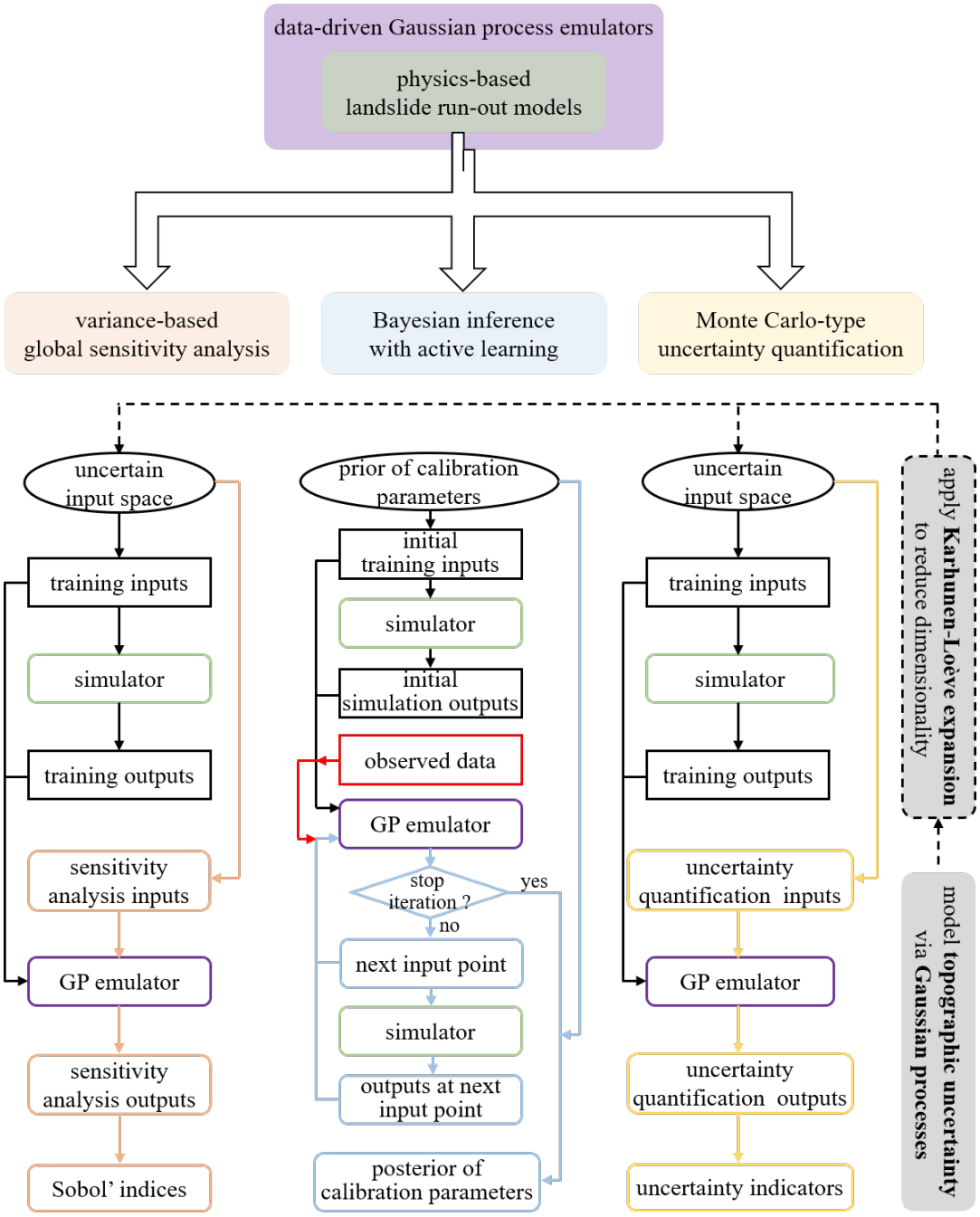

FIGURE 7.1: A schematic illustration of the unified framework developed in this thesis. It enables us to systematically, routinely, and efficiently investigate various uncertainties associated with landslide run-out models.

outputs to the release volume and the two Voellmy rheological parameters was studied in detail. The results show that the proposed emulator-based global sensitivity analysis method can be successfully used to study the relative importance and interactions of input variables in landslide run-out models. The first-order effects of each input variable are found to be broadly in line with results of common one-at-a-time sensitivity analyses in the literature. Moreover, the proposed method allows analyzing interactions between input variables along the full flow path. Such information

cannot be obtained by methods used in current practice.

In chapter 5, a new methodology for efficient parameter calibration of landslide run-out models was presented. It was developed by integrating depth-averaged landslide run-out models, Bayesian inference, Gaussian process emulation, and active learning. The mass flow simulation tool r.avaflow, R package RobustGaSP for robust Gaussian process emulation, and a self-implemented Bayesian active learning scheme were coupled within the unified Python-based framework. The feasibility and efficiency of the proposed method were investigated based on the 2017 Bondo landslide event with synthetic observed data. Furthermore, the impact of different types of observed data was studied using the proposed method. The results prove that the proposed emulator-based Bayesian active learning method is capable of correctly calibrating the rheological parameters. Compared to commonly used probabilistic methods without emulation techniques, the proposed method can greatly improve the computational efficiency by reducing necessary simulation runs from thousands or tens of thousands to a few hundreds. Compared to emulator-based Bayesian inference without active learning, it can better approximate the posterior of rheological parameters with the same computational budget. The results also show that different types of observed data provide different information on rheological parameters and therefore should be combined in the calibration process in order to improve the quality of calibration results.

Chapter 6 was devoted to uncertainty quantification of landslide run-out models. The focus was put on topographic uncertainty which is often neglected in current practice. Two types of geostatistical methods, namely unconditional and conditional stochastic simulations, were introduced and used to study the impact of topographic uncertainty on landslide run-out modeling. The computational workflow including stochastic simulations to generate DEM realizations, landslide run-out modeling, statistical analysis and visualization, and terrain analysis was implemented within the unified framework in a modular way. Based on the 2008 Yu Tung landslide event, a case study was designed and performed. How DEM uncertainty propagates in the results of landslide run-out analyses was discussed in detail. The results show that topographic uncertainty significantly affects landslide run-out modeling depending on how well the underlying flow path is represented. Both the unconditional and conditional stochastic simulation methods are proven to be capable of investigating topographic uncertainty propagation in landslide run-out modeling. Their main difference is that unconditional stochastic simulation can be conducted based on RMSE information only, while conditional stochastic simulation requires the availability of higher-accuracy reference data and is thus more accurate. Moreover, the close relation between the two geostatistical methods and Gaussian processes was revealed, based on which a new method was proposed in order to make Gaussian process emulation also applicable for high-dimensional topographic uncertainty. The new method employs Karhunen–Loève expansion to reduce the dimensionality of topographic uncertainty. After the dimensionality reduction, topographic uncertainty

can be represented by a small number of random variables and can therefore be treated the same as other low-dimensional uncertainty factors within the unified emulator-based framework, such as release areas and volumes, rheological parameters, etc.

7.2 Recommendations for future work

New methodologies were developed in this thesis to better understand the impact of various uncertainties associated with landslide run-out models. A unified Python-based framework was implemented which provides systematical, routine, and efficient tools to address both forward and inverse problems arising from the uncertainties. Case studies were conducted to validate the new methodologies and investigate the various uncertain factors. While the methods and framework have great potentials, much work remains to be done in order to achieve reliable and transparent simulation-based hazard mapping, risk assessment, and mitigation strategy design. Some recommendations for future work are given as follows:

- The Karhunen–Loève expansion-based method proposed in section 6.6 is attractive and promising. It has the potential to reduce the high-dimensional topographic uncertainty to a low-dimensional problem and make Gaussian process emulation also applicable to topographic uncertainty. Its implementation and application are not addressed in this thesis and therefore require future work.

- For simplicity, the depth-averaged landslide run-out model used in the case studies is restricted to a simple form without taking topographic curve effects and entrainment processes into account. Studies in the literature have shown that these factors may impact simulation results and therefore may influence the results of the global sensitivity analysis, Bayesian inference, and uncertainty quantification. Work towards this direction should be conducted in the future.

- The topography can greatly affect simulation results. Whether and to what extent the conclusions drawn from the case studies based on the Bondo site and Yu Tung site can be applied elsewhere should be studied in the future. Case studies based on different types of topography are necessary to reach common and transferable results.

- The methodologies developed in this thesis have only been applied to depth-averaged landslide run-out model with the Voellmy rheology. It would be useful to apply the methodologies also to other rheological models in the future.

- The significant improvement of computational efficiency owing to emulation techniques may also benefit simulation-based early warning systems. How to

integrate emulation techniques with existing early warning systems can be an interesting topic and deserves future study.

Bibliography

Aaron, J., S. McDougall, and N. Nolde (2019). "Two methodologies to calibrate landslide runout models". In: *Landslides* 16, pp. 907–920. DOI: https://doi.org/10.1007/s10346-018-1116-8.

Aaron, J. (2017). "Advancement and calibration of a 3D numerical model for landslide runout analysis". PhD thesis. Vancouver: The University of British Columbia.

Aaron, J. and O. Hungr (2016). "Dynamic simulation of the motion of partially-coherent landslides". In: *Engineering Geology* 205, pp. 1–11. DOI: https://doi.org/10.1016/j.enggeo.2016.02.006.

Aaron, J. and S. McDougall (2019). "Rock avalanche mobility: The role of path material". In: *Engineering Geology* 257, p. 105126. DOI: https://doi.org/10.1016/j.enggeo.2019.05.003.

Abe, K. and K. Konagai (2016). "Numerical simulation for runout process of debris flow using depth-averaged material point method". In: *Soils and Foundations* 56.5, pp. 869–888. DOI: https://doi.org/10.1016/j.sandf.2016.08.011.

AECOM Asia Company Limited (2012). *Detailed study of the 7 June 2008 landslides on the hillshade above Yu Tung Road, Tung Chung*. Tech. rep.

Aleksankina, K., S. Reis, M. Vieno, and M. R. Heal (2019). "Advanced methods for uncertainty assessment and global sensitivity analysis of an Eulerian atmospheric chemistry transport model". In: *Atmospheric Chemistry and Physics* 19.5, pp. 2881–2898. DOI: 10.5194/acp-19-2881-2019.

Álvarez, M. A. and N. D. Lawrence (2011). "Computationally efficient convolved multiple output Gaussian processes". In: *Journal of Machine Learning Research* 12.41, pp. 1459–1500. URL: http://jmlr.org/papers/v12/alvarez11a.html.

Andrianakis, Y and P. Challenor (2009). *Parameter estimation and prediction using Gaussian processes*. Tech. rep. Managing Uncertainty in Complex Models. URL: http://mucm.ac.uk/Pages/Dissemination/TechnicalReports.html.

— (2011). *Parameter estimation for Gaussian process emulators*. Tech. rep. Managing Uncertainty in Complex Models. URL: http://mucm.ac.uk/Pages/Dissemination/TechnicalReports.html.

Archer, G. E. B., A. Saltelli, and I. M. Sobol (1997). "Sensitivity measures, anova-like techniques and the use of bootstrap". In: *Journal of Statistical Computation and Simulation* 58.2, pp. 99–120. DOI: 10.1080/00949659708811825.

Asher, M. J., B. F. W. Croke, A. J. Jakeman, and L. J. M. Peeters (2015). "A review of surrogate models and their application to groundwater modeling". In: *Water Resources Research* 51.8, pp. 5957–5973. DOI: https://doi.org/10.1002/2015WR016967.

Aziz, S. A., B. L. Steward, A. L. Kaleita, and M. Karkee (2012). "Assessing the effects of DEM uncertainty on erosion rate estimation in an agricultural field". In: *Transactions of the ASABE* 55.3, pp. 785–798. DOI: 10.13031/2013.41514.

Ba, S. and V. R. Joseph (2012). "Composite Gaussian process models for emulating expensive functions". In: *The Annals of Applied Statistics* 6.4, pp. 1838–1860. DOI: 10.1214/12-AOAS570.

Barbolini, M., U. Gruber, C. J. Keylock, M. Naaim, and F. Savi (2000). "Application of statistical and hydraulic-continuum dense-snow avalanche models to five real European sites". In: *Cold Regions Science and Technology* 31.2, pp. 133–149. DOI: https://doi.org/10.1016/S0165-232X(00)00008-2.

Bayarri, M. J., J. O. Berger, R. Paulo, J. Sacks, J. A. Cafeo, J. Cavendish, C.-H. Lin, and J. Tu (2007). "A framework for validation of computer models". In: *Technometrics* 49.2, pp. 138–154. DOI: 10.1198/004017007000000092.

Beguería, S., T. W. J. Van Asch, J. P. Malet, and S. Gröndahl (2009). "A GIS-based numerical model for simulating the kinematics of mud and debris flows over complex terrain". In: *Natural Hazards and Earth System Sciences* 9.6, pp. 1897–1909. DOI: 10.5194/nhess-9-1897-2009.

Berry, P. A. M., J. D. Garlick, and R. G. Smith (2007). "Near-global validation of the SRTM DEM using satellite radar altimetry". In: *Remote Sensing of Environment* 106, pp. 17–27. DOI: 10.1016/j.rse.2006.07.011.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. New York: Springer. ISBN: 978-0-387-31073-2.

Bolkas, D., G. Fotopoulos, A. Braun, and I. N. Tziavos (2016). "Assessing digital elevation model uncertainty using GPS survey data". In: *Journal of Surveying Engineering* 142.3, p. 04016001. DOI: 10.1061/(ASCE)SU.1943-5428.0000169.

Borstad, C. P. and D. M. McClung (2009). "Sensitivity analyses in snow avalanche dynamics modeling and implications when modeling extreme events". In: *Canadian Geotechnical Journal* 46.9, pp. 1024–1033. DOI: https://doi.org/10.1139/T09-042.

Bounceur, N., M. Crucifix, and R. D. Wilkinson (2015). "Global sensitivity analysis of the climate–vegetation system to astronomical forcing: an emulator-based approach". In: *Earth System Dynamics* 6.1, pp. 205–224. DOI: 10.5194/esd-6-205-2015.

Brezzi, L., F. Gabrieli, G. Marcato, M. Pastor, and S. Cola (2016). "A new data assimilation procedure to develop a debris flow run-out model". In: *Landslides* 13, pp. 1083–1096. DOI: 10.1007/s10346-015-0625-y.

Calvello, M., S. Cuomo, and P. Ghasemi (2017). "The role of observations in the inverse analysis of landslide propagation". In: *Computers and Geotechnics* 92, pp. 11–21. DOI: https://doi.org/10.1016/j.compgeo.2017.07.011.

Chen, H., G. B. Crosta, and C. F. Lee (2006). "Erosional effects on runout of fast landslides, debris flows and avalanches: a numerical investigation". In: *Géotechnique* 56.5, pp. 305–322. DOI: 10.1680/geot.2006.56.5.305.

Chen, H. and C. F. Lee (2000). "Numerical simulation of debris flows". In: *Canadian Geotechnical Journal* 37.1, pp. 146–160. DOI: 10.1139/t99-089.

— (2004). "Geohazards of slope mass movement and its prevention in Hong Kong". In: *Engineering Geology* 76, pp. 3–25. DOI: 10.1016/j.enggeo.2004.06.003.

Christen, M., J. Kowalski, and P. Bartelt (2010). "RAMMS: Numerical simulation of dense snow avalanches in three-dimensional terrain". In: *Cold Regions Science and Technology* 63.1, pp. 1–14. DOI: https://doi.org/10.1016/j.coldregions.2010.04.005.

Courty, L. G., J. C. Soriano-Monzalvo, and A. Pedrozo-Acuna (2019). "Evaluation of open-access global digital elevation models (AW3D30, SRTM, and ASTER) for flood modelling purposes". In: *Journal of Flood Risk Management* 12.S1, e12550. DOI: 10.1111/jfr3.12550.

Cranmer, K., J. Brehmer, and G. Louppe (2020). "The frontier of simulation-based inference". In: *Proceedings of the National Academy of Sciences* 117.48, pp. 30055–30062. DOI: 10.1073/pnas.1912789117.

Cressie, N. A. C. (1993). *Statistics for spatial data*. John Wiley & Sons, Inc. ISBN: 97804710-02550. DOI: 10.1002/9781119115151.

Crosta, G. B., S. Imposimato, and D. G. Roddeman (2003). "Numerical modelling of large landslides stability and runout". In: *Natural Hazards and Earth System Sciences* 3.6, pp. 523–538. DOI: 10.5194/nhess-3-523-2003.

Currin, C., T. Mitchell, M. Morris, and D. Ylvisaker (1991). "Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments". In: *Journal of the American Statistical Association* 86.416, pp. 953–963. DOI: 10.1080/01621459.1991.10475138.

DATA.GOV.HK (2019). *5 m resolution digital terrain model of the Hong Kong Special Administrative Region*. https://www.landsd.gov.hk/mapping/en/download/psi/opendata.htm.

De Blasio, F. V. (2011). *Introduction to the physics of landslides. Lecture Notes on the Dynamics of Mass Wasting*. New York: Springer. DOI: 10.1007/978-94-007-1122-8.

Denlinger, R. P. and R. M. Iverson (2001). "Flow of variably fluidized granular masses across three-dimensional terrain: 2. Numerical predictions and experimental tests". In: *Journal of Geophysical Research: Solid Earth* 106.B1, pp. 553–566. DOI: https://doi.org/10.1029/2000JB900330.

Dilley, M., R. S. Chen, U. Deichmann, A. L. Lerner-Lam, and M. Arnold (2005). *Natural disaster hotspots: A global risk analysis*. Tech. rep. Washington, DC: The World Bank. URL: http://hdl.handle.net/10986/7376.

Elkhrachy, I. (2018). "Vertical accuracy assessment for SRTM and ASTER digital elevation models: a case study of Najran city, Saudi Arabia". In: *Ain Shams Engineering Journal* 9.4, pp. 1807–1817. DOI: 10.1016/j.asej.2017.01.007.

Fathani, T. F., D. Legono, and M. A. Alfath (2017). "Sensitivity analysis of depth-integrated numerical models for estimating landslide movement". In: *Journal of Disaster Research* 12.3, pp. 607–616. DOI: 10.20965/jdr.2017.p0607.

Fer, I., R. Kelly, P. R. Moorcroft, A. D. Richardson, E. M. Cowdery, and M. C. Dietze (2018). "Linking big models to big data: efficient ecosystem model calibration through Bayesian model emulation". In: *Biogeosciences* 15.19, pp. 5801–5830. DOI: 10.5194/bg-15-5801-2018.

Fischer, J.-T., A. Kofler, W. Fellin, M. Granig, and K. Kleemayr (2015). "Multivariate parameter optimization for computational snow avalanche simulation". In: *Journal of Glaciology* 61.229, pp. 875–888. DOI: 10.3189/2015JoG14J168.

Fischer, J.-T., J. Kowalski, and S. P. Pudasaini (2012). "Topographic curvature effects in applied avalanche modeling". In: *Cold Regions Science and Technology* 74-75, pp. 21–30. DOI: https://doi.org/10.1016/j.coldregions.2012.01.005.

Fisher, P. F. and N. J. Tate (2006). "Causes and consequences of error in digital elevation models". In: *Progress in Physical Geography: Earth and Environment* 30.4, pp. 467–489. DOI: 10.1191/0309133306pp492ra.

Foreman-Mackey, D., D. W. Hogg, D. Lang, and J. Goodman (2013). "emcee: The MCMC Hammer". In: *Publications of the Astronomical Society of the Pacific* 125.925, pp. 306–312. DOI: 10.1086/670067.

Frank, F., B. W. McArdell, C. Huggel, and A. Vieli (2015). "The importance of entrainment and bulking on debris flow runout modeling: examples from the Swiss Alps". In: *Natural Hazards and Earth System Sciences* 15.11, pp. 2569–2583. DOI: 10.5194/nhess-15-2569-2015.

Frey, H., C. Huggel, Y. Bühler, D. Buis, M. D. Burga, W. Choquevilca, F. Fernandez, J. G. Hernández, C. Giráldez, E. Loarte, P. Masias, C. Portocarrero, L. Vicuña, and M. Walser (2016). "A robust debris-flow and GLOF risk management strategy for a data-scarce catchment in Santa Teresa, Peru". In: *Landslides* 13, pp. 1493–1507. DOI: 10.1007/s10346-015-0669-z.

Froude, M. J. and D. N. Petley (2018). "Global fatal landslide occurrence from 2004 to 2016". In: *Natural Hazards and Earth System Sciences* 18, pp. 2161–2181. DOI: 10.5194/nhess-18-2161-2018.

Garbuno-Inigo, A., F. A. DiazDelaO, and K. M. Zuev (2016). "Gaussian process hyper-parameter estimation using parallel asymptotically independent Markov sampling". In: *Computational Statistics and Data Analysis* 103, pp. 367–383. DOI: https://doi.org/10.1016/j.csda.2016.05.019.

García-Rodríguez, M. J. and J. A. Malpica (2010). "Assessment of earthquake-triggered landslide susceptibility in El Salvador based on an Artificial Neural Network model". In: *Natural Hazards and Earth System Sciences* 10.6, pp. 1307–1315. DOI: 10.5194/nhess-10-1307-2010.

Gariano, S. L. and F. Guzzetti (2016). "Landslides in a changing climate". In: *Earth-Science Reviews* 162, pp. 227–252. DOI: https://doi.org/10.1016/j.earscirev.2016.08.011.

Garres-Díaz, J., E. D. Fernández-Nieto, A. Mangeney, and T. M. de Luna (2021). "A weakly non-hydrostatic shallow model for dry granular flows". In: *Journal of Scientific Computing* 86.25. DOI: https://doi.org/10.1007/s10915-020-01377-9.

Gelman, A., J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin (2013). *Bayesian data analysis. Third Edition*. Boca Raton: CRC Press. ISBN: 978-1-4398-9820-8.

George, D. L. and R. M. Iverson (2014). "A depth-averaged debris-flow model that includes the effects of evolving dilatancy. II. Numerical predictions and experimental tests". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 470, p. 20130820. DOI: 10.1098/rspa.2013.0820.

Gershman, S. J. and D. M. Blei (2012). "A tutorial on Bayesian nonparametric models". In: *Journal of Mathematical Psychology* 56.1, pp. 1–12. DOI: https://doi.org/10.1016/j.jmp.2011.08.004.

Girard, S., V. Mallet, I. Korsakissok, and A. Mathieu (2016). "Emulation and Sobol' sensitivity analysis of an atmospheric dispersion model applied to the Fukushima nuclear accident". In: *Journal of Geophysical Research: Atmospheres* 121.7, pp. 3484–3496. DOI: 10.1002/2015JD023993.

Gonga-Saholiariliva, N., Y. Gunnell, C. Petit, and C. Mering (2011). "Techniques for quantifying the accuracy of gridded elevation models and for mapping uncertainty in digital terrain analysis". In: *Progress in Physical Geography: Earth and Environment* 35.6, pp. 739–764. DOI: 10.1177/0309133311409086.

Goodman, J. and J. Weare (2010). "Ensemble samplers with affine invariance". In: *Communications in Applied Mathematics and Computational Science* 5.1, pp. 65–80. DOI: 10.2140/camcos.2010.5.65.

Goovaerts, P. (1997). *Geostatistics for natural resources evaluation*. New York: Oxford University Press. ISBN: 0-19-511538-4.

Gu, M. Y. and J. O. Berger (2016). "Parallel partial Gaussian process emulation for computer models with massive output". In: *Annals of Applied Statistics* 10.3, pp. 1317–1347. DOI: 10.1214/16-AOAS934.

Gu, M. Y., J. Palomo, and J. O. Berger (2019). "RobustGaSP: Robust Gaussian stochastic process emulation in R". In: *The R Journal* 11.1, pp. 112–136. DOI: 10.32614/RJ-2019-011.

Gu, M. Y. and L. Wang (2018). "Scaled Gaussian stochastic process for computer model calibration and prediction". In: *SIAM/ASA Journal on Uncertainty Quantification* 6.4, pp. 1555–1583. DOI: 10.1137/17M1159890.

Gu, M. Y., X. J. Wang, and J. O. Berger (2018). "Robust Gaussian stochastic process emulation". In: *Annals of Statistics* 46.6A, pp. 3038–3066. DOI: 10.1214/17-AOS1648.

Handcock, M. S. and M. L. Stein (1993). "A Bayesian analysis of kriging". In: *Technometrics* 35.4, pp. 403–410. DOI: 10.1080/00401706.1993.10485354.

Hawker, L., P. Bates, J. Neal, and J. Rougier (2018). "Perspectives on digital elevation model (DEM) simulation for flood modeling in the absence of a high-accuracy open assess global DEM". In: *Frontiers in Earth Science* 6, p. 233. DOI: 10.3389/feart.2018.00233.

Hengl, T., S. Gruber, and D. P. Shrestha (2004). "Reduction of errors in digital terrain parameters used in soil-landscape modeling". In: *International Journal of Applied Earth Observation and Geoinformation* 5.2, pp. 97–112. DOI: 10.1016/j.jag.2004.01.006.

Heredia, M. B., N. Eckert, C. Prieur, and E. Thibert (2020). "Bayesian calibration of an avalanche model from autocorrelated measurements along the flow: application to velocities extracted from photogrammetric images". In: *Journal of Glaciology* 66.257, 373—385. DOI: 10.1017/jog.2020.11.

Hergarten, S. and J. Robl (2015). "Modelling rapid mass movements using the shallow water equations in Cartesian coordinates". In: *Natural Hazards and Earth System Sciences* 15.3, pp. 671–685. DOI: 10.5194/nhess-15-671-2015.

Herman, J. and W. Usher (2017). "SALib: An open-source Python library for Sensitivity Analysis". In: *The Journal of Open Source Software* 2.9. DOI: 10.21105/joss.00097.

Higdon, D., J. Gattiker, B. Williams, and M. Rightley (2008). "Computer model calibration using high-dimensional output". In: *Journal of the American Statistical Association* 103.482, pp. 570–583. DOI: 10.1198/016214507000000888.

Hofert, M. (2013). "On sampling from the multivariate t distribution". In: *The R Journal* 5, pp. 130–137. DOI: 10.32614/RJ-2013-033.

Hofton, M., R. Dubayah, J. B. Blair, and D. Rabine (2006). "Validation of SRTM elevations over vegetated and non-vegetated terrain using medium footprint LiDAR". In: *Photogrammetric Engineering & Remote Sensing* 72.3, pp. 279–285. DOI: 10.14358/PERS.72.3.279.

Holmes, K. W., O. A. Chadwick, and P. C. Kyriakidis (2000). "Error in a USGS 30-meter digital elevation model and its impact on terrain modeling". In: *Journal of Hydrology* 233.1-4, pp. 154–173. DOI: 10.1016/S0022-1694(00)00229-8.

Hungr, O. (1995). "A model for the runout analysis of rapid flow slides, debris flows, and avalanches". In: *Canadian Geotechnical Journal* 32, pp. 610–623. DOI: 10.1139/t95-063.

— (2009). "Numerical modelling of the motion of rapid, flow-like landslides for hazard assessment". In: *KSCE Journal of Civil Engineering* 13.4, pp. 281–287. DOI: 10.1007/s12205-009-0281-7.

Hungr, O., N. Morgenstern, and H. N. Wong (2007). "Review of benchmarking exercise on landslide debris runout and mobility modelling". In: *Proceedings of the 2007 International Forum on Landslide Disaster Management*. Ed. by K. Ho and V. Li, pp. 755–812. ISBN: 978-962-7619-30-7.

Hungr, O. and S. McDougall (2009). "Two numerical models for landslide dynamic analysis". In: *Computers and Geosciences* 35.5, pp. 978–992. DOI: https://doi.org/10.1016/j.cageo.2007.12.003.

Hussin, H. Y., B. Quan Luna, C. J. van Westen, M. Christen, J. P. Malet, and T. W. J. van Asch (2012). "Parameterization of a numerical 2-D debris flow model with entrainment: a case study of the Faucon catchment, Southern French Alps". In: *Natural Hazards and Earth System Sciences* 12.10, pp. 3075–3090. DOI: 10.5194/nhess-12-3075-2012.

Iverson, R. M. (2012). "Elementary theory of bed-sediment entrainment by debris flows and avalanches". In: *Journal of Geophysical Research: Earth Surface* 117.F3. DOI: https://doi.org/10.1029/2011JF002189.

Iverson, R. M. and D. L. George (2014). "A depth-averaged debris-flow model that includes the effects of evolving dilatancy. I. Physical basis". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 470, p. 20130819. DOI: 10.1098/rspa.2013.0819.

Janon, A., M. Nodet, and C. Prieur (2014). "Uncertainties assessment in global sensitivity indices estimation from metamodels". In: *International Journal for Uncertainty Quantification* 4, pp. 21–36. DOI: 10.1615/Int.J.UncertaintyQuantification.2012004291.

Järvenpää, M., M. U. Gutmann, A. Vehtari, and P. Marttinen (2021). "Parallel Gaussian process surrogate Bayesian inference with noisy likelihood evaluations". In: *Bayesian Analysis* 16.1, pp. 147–178. DOI: 10.1214/20-BA1200.

Kandasamy, K., J. Schneider, and B. Póczos (2017). "Query efficient posterior estimation in scientific experiments via Bayesian active learning". In: *Artificial Intelligence* 243, pp. 45–56. DOI: https://doi.org/10.1016/j.artint.2016.11.002.

Kennedy, M. C. and A. O'Hagan (2001). "Bayesian calibration of computer models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3, pp. 425–464. DOI: https://doi.org/10.1111/1467-9868.00294.

Kiczko, A. and D. Miroslaw-Swiatek (2018). "Impact of uncertainty of floodplain digital terrain model on 1D hydrodynamic flow calculation". In: *Water* 10.10, p. 1308. DOI: 10.3390/w10101308.

Krige, D. G. (1951). "A statistical approach to some basic mine valuation problems on the Witwatersrand". In: *Journal of the Chemical Metallurgical and Mining Society of South Africa* 52.6, pp. 119–139.

Krishnan, S., C. J. Crosby, V. Nandigam, M. Phan, C. Cowart, C. Baru, and R. J. Arrowsmith (2011). "OpenTopography: a services oriented architecture for community access to LiDAR topography". In: *ACM International Conference Proceeding Series*, p. 7. DOI: 10.1145/1999320.1999327.

Kwan, J. S. H. and H. W. Sun (2007). "Benchmarking exercise on landslide mobility modelling - runout analyses using 3dDMM". In: *Proceedings of the 2007 International Forum on Landslide Disaster Management*. Ed. by K Ho and V Li, pp. 945–966. ISBN: 978-962-7619-30-7.

Le Gratiet, L., C. Cannamela, and B. Iooss (2014). "A Bayesian approach for global sensitivity analysis of (multifidelity) computer codes". In: *SIAM/ASA Journal on Uncertainty Quantification* 2.1, pp. 336–363. DOI: 10.1137/130926869.

Lee, L. A., K. S. Carslaw, K. J. Pringle, and G. W. Mann (2012). "Mapping the uncertainty in global CCN using emulation". In: *Atmospheric Chemistry and Physics* 12.20, pp. 9739–9751. DOI: 10.5194/acp-12-9739-2012.

Lee, L. A., K. S. Carslaw, K. J. Pringle, G. W. Mann, and D. V. Spracklen (2011). "Emulation of a complex global aerosol model to quantify sensitivity to uncertain parameters". In: *Atmospheric Chemistry and Physics* 11.23, pp. 12253–12273. DOI: 10.5194/acp-11-12253-2011.

Li, Y., X. Wang, and H. Mao (2020). "Influence of human activity on landslide susceptibility development in the Three Gorges area". In: *Natural Hazards* 104, pp. 2115–2151. DOI: https://doi.org/10.1007/s11069-020-04264-6.

Lindsay, J. B. (2018). *WhiteboxTools user manual*. Geomorphometry and Hydrogeomatics Research Group.

Liu, Y., J. F. Li, S. Y. Sun, and B. Yu (2019). "Advances in Gaussian random field generation: a review". In: *Computational Geosciences* 23, pp. 1011–1047. DOI: https://doi.org/10.1007/s10596-019-09867-y.

Lucas, A., A. Mangeney, and J. Ampuero (2014). "Frictional velocity-weakening in landslides on Earth and on other planetary bodies". In: *Nature Communications* 5.3417. DOI: 10.1038/ncomms4417.

Maatouk, H. and X. Bay (2017). "Gaussian process emulators for computer experiments with inequality constraints". In: *Mathematical Geosciences* 49, pp. 557–582. DOI: https://doi.org/10.1007/s11004-017-9673-2.

Mangeney, A., P. Heinrich, and R. Roche (2000). "Analytical solution for testing debris avalanche numerical models". In: *Pure and Applied Geophysics* 157, pp. 1081–1096. DOI: https://doi.org/10.1007/s000240050018.

Mangeney-Castelnau, A., J. P. Vilotte, M. O. Bristeau, B. Perthame, F. Bouchut, C. Simeoni, and S. Yerneni (2003). "Numerical modeling of avalanches based on Saint Venant equations using a kinetic scheme". In: *Journal of Geophysical Research: Solid Earth* 108.B11. DOI: https://doi.org/10.1029/2002JB002024.

Marrel, A., B. Iooss, B. Laurent, and O. Roustant (2009). "Calculations of Sobol indices for the Gaussian process metamodel". In: *Reliability Engineering and System Safety* 94.3, pp. 742–751. DOI: https://doi.org/10.1016/j.ress.2008.07.008.

Matheron, G. (1963). "Principles of geostatistics". In: *Economic Geology* 58.8, pp. 1246–1266. DOI: 10.2113/gsecongeo.58.8.1246.

McDougall, S. (2017). "2014 Canadian Geotechnical Colloquium: landslide runout analysis – current practice and challenges". In: *Canadian Geotechnical Journal* 54.5, pp. 605–620. DOI: 10.1139/cgj-2016-0104.

McMillan, H. and M. Clark (2009). "Rainfall-runoff model calibration using informal likelihood measures within a Markov chain Monte Carlo sampling scheme". In: *Water Resources Research* 45.4. DOI: https://doi.org/10.1029/2008WR007288.

Medina, V., M. Hürlimann, and A. Bateman (2008). "Application of FLATModel, a 2D finite volume code, to debris flows in the northeastern part of the Iberian Peninsula". In: *Landslides* 5, pp. 127–142. DOI: 10.1007/s10346-007-0102-3.

Mergili, M., J. T. Fischer, J. Krenn, and S. P. Pudasaini (2017). "r.avaflow v1, an advanced open-source computational framework for the propagation and interaction of two-phase mass flows". In: *Geoscientific Model Development* 10.2, pp. 553–569. DOI: 10.5194/gmd-10-553-2017.

Mergili, M., M. Jaboyedoff, J. Pullarello, and S. P. Pudasaini (2020). "Back calculation of the 2017 Piz Cengalo–Bondo landslide cascade with r.avaflow: what we can do and what we can learn". In: *Natural Hazards and Earth System Sciences* 20.2, pp. 505–520. DOI: 10.5194/nhess-20-505-2020.

Mergili, M., K. Schratz, A. Ostermann, and W. Fellin (2012). "Physically-based modelling of granular flows with open source GIS". In: *Natural Hazards and Earth System Sciences* 12.1, pp. 187–200. DOI: 10.5194/nhess-12-187-2012.

Miller, C. L. and R. A. Laflamme (1958). "The digital terrain model – theory & application". In: *Photogrammetric Engineering* 24.3, pp. 433–442.

Moawad, M. B. and A. O. A. EI Aziz (2018). "Assessment of remotely sensed digital elevation models (DEMs) compared with DGPS elevation data and its influence on topographic attributes". In: *Advances in Remote Sensing* 7.2, pp. 144–162. DOI: 10.4236/ars.2018.72010.

Mohammadi, H., P. Challenor, and M. Goodfellow (2019). "Emulating dynamic non-linear simulators using Gaussian processes". In: *Computational Statistics and Data Analysis* 139, pp. 178–196. DOI: https://doi.org/10.1016/j.csda.2019.05.006.

Moretti, L., K. Allstadt, A. Mangeney, Y. Capdeville, E. Stutzmann, and F. Bouchut (2015). "Numerical modeling of the Mount Meager landslide constrained by its force history derived from seismic data". In: *Journal of Geophysical Research: Solid Earth* 120.4, pp. 2579–2599. DOI: https://doi.org/10.1002/2014JB011426.

Moretti, L., A. Mangeney, F. Walter, Y. Capdeville, T. Bodin, E. Stutzmann, and A. Le Friant (2020). "Constraining landslide characteristics with Bayesian inversion of field and seismic data". In: *Geophysical Journal International* 221.2, pp. 1341–1348. DOI: 10.1093/gji/ggaa056.

Mouratidis, A. and D. Ampatzidis (2019). "European digital elevation model validation against extensive global navigation satellite systems data and comparison with SRTM DEM and ASTER GDEM in central Macedonia (Greece)". In: *International Journal of Geo-Information* 8.3, p. 108. DOI: 10.3390/ijgi8030108.

Naef, D., D. Rickenmann, P. Rutschmann, and B. W. McArdell (2006). "Comparison of flow resistance relations for debris flows using a one-dimensional finite element simulation model". In: *Natural Hazards and Earth System Sciences* 6.1, pp. 155–165. DOI: 10.5194/nhess-6-155-2006.

Navarro, M., O. P. Le Maître, I. Hoteit, D. L. George, K. T. Mandli, and O. M. Knio (2018). "Surrogate-based parameter inference in debris flow model". In: *Computational Geosciences* 22, pp. 1447–1463. DOI: https://doi.org/10.1007/s10596-018-9765-1.

Oakley, J. E. and A. O'Hagan (2004). "Probabilistic sensitivity analysis of complex models: a Bayesian approach". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 66.3, pp. 751–769. DOI: 10.1111/j.1467-9868.2004.05304.x.

Oakley, J. E. and B. D. Youngman (2017). "Calibration of stochastic computer simulators using likelihood emulation". In: *Technometrics* 59.1, pp. 80–92. DOI: 10.1080/00401706.2015.1125391.

O'Brien, J. S., P. Y. Julien, and W. T. Fullerton (1993). "Two dimensional water flood and mudflow simulation". In: *Journal of Hydraulic Engineering* 119.2, pp. 244–261. DOI: 10.1061/(ASCE)0733-9429(1993)119:2(244).

O'Hagan, A. (2006). "Bayesian analysis of computer code outputs: A tutorial". In: *Reliability Engineering and System Safety* 91.10, pp. 1290–1300. DOI: https://doi.org/10.1016/j.ress.2005.11.025.

Oksanen, J. (2003). "Tracing the gross errors of DEM-visualization techniques for preliminary quality analysis". In: *Proceedings of the 21st International Cartographic Conference 'Cartographic Renaissance'*, pp. 2410–2416. ISBN: 0-958-46093-0.

— (2006). "Digital elevation model error in terrain analysis". PhD thesis. Helsinki: University of Helsinki.

Omre, H. (1987). "Bayesian kriging–Merging observations and qualified guesses in kriging". In: *Mathematical Geology* 19, pp. 25–39. DOI: https://doi.org/10.1007/BF01275432.

Ouyang, C. J., S. M. He, Q. Xu, Y. Luo, and W. C. Zhang (2013). "A MacCormack-TVD finite difference method to simulate the mass flow in mountainous terrain with variable computational domain". In: *Computers & Geosciences* 52, pp. 1–10. DOI: https://doi.org/10.1016/j.cageo.2012.08.024.

Pakoksung, K. and M. Takagi (2016). "Digital elevation models on accuracy validation and bias correction in vertical". In: *Modeling Earth Systems and Environment* 2.1, p. 11. DOI: 10.1007/s40808-015-0069-3.

Pastor, M., T. Blanc, D. Manzanal, V. Drempetic, M. J. Pastor, M. Sánchez, G. Crosta, S. Imposimato, D. Roddeman, E. Foester, H. Kobayashi, M. Delattre, and D. Issler (2012). "Landslide runout: Review of analytical/empirical models for subaerial slides, submarine slides and snow avalanche. Numerical modelling. Software tools, material models, validation and benchmarking for selected case studies". In: *SafeLand Deliverable D1.7 (Revision 2)*. URL: https://www.ngi.no/eng/Projects/SafeLand/#Reports-and-publications.

Pastor, M., B. Haddad, G. Sorbino, S. Cuomo, and V. Drempetic (2009). "A depth-integrated, coupled SPH model for flow-like landslides and related phenomena". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 33.2, pp. 143–172. DOI: https://doi.org/10.1002/nag.705.

Pastor, M., K. Soga, S. McDougall, and J. S. H. Kwan (2018). "Review of benchmarking exercise on landslide runout analysis 2018". In: *Proceedings of the Second JTC1 Workshop on Triggering and Propagation of Rapid Flow-like Landslides*. Ed. by K. Ho, A. Leung, J. Kwan, R. Koo, and R. Law. The Hong Kong Geotechnical Society, pp. 281–323. ISBN: 978-988-74066-0-0.

Patel, A., S. K. Katiyar, and V. Prasad (2016). "Performances evaluation of different open source DEM using differential global positioning system (DGPS)". In: *The Egyptian Journal of Remote Sensing and Space Science* 19.1, pp. 7–16. DOI: 10.1016/j.ejrs.2015.12.004.

Pirulli, M. and A. Mangeney (2008). "Results of back-analysis of the propagation of rock avalanches as a function of the assumed rheology". In: *Rock Mechanics and Rock Engineering* 41, pp. 59–84. DOI: https://doi.org/10.1007/s00603-007-0143-x.

Pitman, E. B., C. C. Nichita, A. Patra, A. Bauer, M. Sheridan, and M. Bursik (2003). "Computing granular avalanches and landslides". In: *Physics of Fluids* 15.12, pp. 3638–3646. DOI: 10.1063/1.1614253.

Poisel, R. and A. Preh (2008). "3D landslide runout modelling using the Particle Flow Code PFC3D". In: *Proceedings of the 10th International Symposium on Landslides and Engineered Slopes*. Ed. by Z. Chen, J. Zhang, Z. Li, F. Wu, and K Ho, pp. 873–879. ISBN: 978-0-415-41196-7.

Pudasaini, S. P. and M. Mergili (2019). "A multi-phase mass flow model". In: *Journal of Geophysical Research: Earth Surface* 124.12, pp. 2920–2942. DOI: 10.1029/2019JF005204.

Qin, C. Z., L. L. Bao, A. X. Zhu, R. X. Wang, and X. M. Hu (2013). "Uncertainty due to DEM error in landslide susceptibility mapping". In: *International Journal of Geographical Information Science* 27.7, pp. 1364–1380. DOI: 10.1080/13658816.2013.770515.

Quan Luna, B., J. Cepeda, A. Stumpf, C. J. van Westen, A. Remaître, J. Malet, and T. W. J. van Asch (2013). "Analysis and uncertainty quantification of dynamic run-out model parameters for landslides". In: *Landslide Science and Practice: Volume 3: Spatial Analysis and Modelling*. Ed. by C. Margottini, P. Canuti, and K. Sassa. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 315–318. ISBN: 978-3-642-31310-3. DOI: 10.1007/978-3-642-31310-3_42.

Raaflaub, L. D. and M. J. Collins (2006). "The effect of error in gridded digital elevation models on the estimation of topographic parameters". In: *Environmental Modelling & Software* 21.5, pp. 710–732. DOI: 10.1016/j.envsoft.2005.02.003.

Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian processes for machine learning*. Cambridge, Massachusetts: MIT Press. ISBN: 026218253X.

Rauter, M., A. Kofler, A. Huber, and W. Fellin (2018). "faSavageHutterFOAM 1.0: depth-integrated simulation of dense snow avalanches on natural terrain with OpenFOAM". In: *Geoscientific Model Development* 11.7, pp. 2923–2939. DOI: 10.5194/gmd-11-2923-2018.

Razavi, S., B. A. Tolson, and D. H. Burn (2012). "Review of surrogate modeling in water resources". In: *Water Resources Research* 48.7. DOI: https://doi.org/10.1029/2011WR011527.

Remy, N., A. Boucher, and J. B. Wu (2009). *Applied geostatistics with SGeMS*. Cambridge University Press. DOI: https://doi.org/10.1017/CBO9781139150019.

Rodriguez, E., C. S. Morris, and J. E. Belz (2006). "A global assessment of the SRTM performance". In: *Photogrammetric Engineering & Remote Sensing* 72.3, pp. 249–260. DOI: 10.14358/PERS.72.3.249.

Rohmer, J. and E. Foerster (2011). "Global sensitivity analysis of large-scale numerical landslide models based on Gaussian-process meta-modeling". In: *Computers and Geosciences* 37.7, pp. 917–927. DOI: https://doi.org/10.1016/j.cageo.2011.02.020.

Rougier, J. (2008). "Efficient emulators for multivariate deterministic functions". In: *Journal of Computational and Graphical Statistics* 17.4, pp. 827–843. DOI: 10.1198/106186008X384032.

Rougier, J., S. Guillas, A. Maute, and A. D. Richmond (2009). "Expert knowledge and multivariate emulation: The thermosphere–ionosphere electrodynamics general circulation model (TIE-GCM)". In: *Technometrics* 51.4, pp. 414–424. DOI: 10.1198/TECH.2009.07123.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn (1989). "Design and analysis of computer experiments". In: *Statistical Science* 4.4, pp. 409–423. DOI: 10.1214/ss/1177012413.

Saltelli, A., M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Taranola (2008). "Variance-based methods". In: *Global sensitivity analysis. The primer*. John Wiley and Sons, Ltd. Chap. 4, pp. 155–182. ISBN: 9780470725184. DOI: 10.1002/9780470725184.ch4.

Saltelli, A. (2002). "Making best use of model evaluations to compute sensitivity indices". In: *Computer Physics Communications* 145.2, pp. 280–297. DOI: https://doi.org/10.1016/S0010-4655(02)00280-1.

Saltelli, A., P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola (2010). "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index". In: *Computer Physics Communications* 181.2, pp. 259–270. DOI: https://doi.org/10.1016/j.cpc.2009.09.018.

Savage, S. and K. Hutter (1989). "The motion of a finite mass of granular material down a rough incline". In: *Journal of Fluid Mechanics* 199, pp. 177–215. DOI: 10.1017/S0022112089000340.

Schraml, K., B. Thomschitz, B. W. McArdell, C. Graf, and R. Kaitna (2015). "Modeling debris-flow runout patterns on two alpine fans with different dynamic simulation models". In: *Natural Hazards and Earth System Sciences* 15.7, pp. 1483–1492. DOI: 10.5194/nhess-15-1483-2015.

Schulz, E., M. Speekenbrink, and A. Krause (2018). "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions". In: *Journal of Mathematical Psychology* 85, pp. 1–16. DOI: https://doi.org/10.1016/j.jmp.2018.03.001.

Shan, W., Z. Hu, Y. Guo, C. Zhang, C. Wang, H. Jiang, Y. Liu, and J. Xiao (2015). "The impact of climate change on landslides in southeastern of high-latitude permafrost regions of China". In: *Frontiers in Earth Science* 3, p. 7. DOI: 10.3389/feart.2015.00007.

Sinsbeck, M. and W. Nowak (2017). "Sequential design of computer experiments for the solution of Bayesian inverse problems". In: *SIAM/ASA Journal on Uncertainty Quantification* 5.1, pp. 640–664. DOI: 10.1137/15M1047659.

Skilodimou, H. D., G. D. Bathrellos, E. Koskeridou, K. Soukis, and D. Rozos (2018). "Physical and anthropogenic factors related to landslide activity in the northern Peloponnese, Greece". In: *Land* 7.3, p. 85. DOI: 10.3390/land7030085.

Sobol', I. M. (1993). "Sensitivity analysis for nonlinear mathematical models". In: *Mathematical Modelling and Computational Experiment* 1, pp. 407–414.

— (2001). "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates". In: *Mathematics and Computers in Simulation* 55.1, pp. 271–280. DOI: https://doi.org/10.1016/S0378-4754(00)00270-6.

Sosio, R., G. B. Crosta, and O. Hungr (2008). "Complete dynamic modeling calibration for the Thurwieser rock avalanche (Italian Central Alps)". In: *Engineering Geology* 100.1, pp. 11–26. DOI: https://doi.org/10.1016/j.enggeo.2008.02.012.

Spiller, E. T., M. J. Bayarri, J. O. Berger, E. S. Calder, A. K. Patra, E. B. Pitman, and R. L. Wolpert (2014). "Automating emulator construction for geophysical hazard maps". In: *SIAM/ASA Journal on Uncertainty Quantification* 2.1, pp. 126–152. DOI: 10.1137/120899285.

Stefanescu, E. R., M. Bursik, G. Cordoba, K. Dalbey, M. D. Jones, A. K. Patra, D. C. Pieri, E. B. Pitman, and M. F. Sheridan (2012). "Digital elevation model uncertainty and hazard analysis using a geophysical flow model". In: *Proceedings of the Royal Society A* 468.2142, pp. 1543–1563. DOI: 10.1098/rspa.2011.0711.

Stein, M. L. (1999). *Interpolation of spatial data. Some Theory for Kriging*. New York: Springer-Verlag New York, Inc. ISBN: 978-1-4612-1494-6.

Stoffel, M., D. Tiranti, and C. Huggel (2014). "Climate change impacts on mass movements – Case studies from the European Alps". In: *Science of The Total Environment* 493, pp. 1255–1266. DOI: https://doi.org/10.1016/j.scitotenv.2014.02.102.

Sun, X. P., P. Zeng, T. B. Li, S. Wang, R. Jimenez, X. D. Feng, and Q. Xu (2021). "From probabilistic back analyses to probabilistic run-out predictions of landslides: A case study of Heifangtai terrace, Gansu Province, China". In: *Engineering Geology* 280, p. 105950. DOI: https://doi.org/10.1016/j.enggeo.2020.105950.

Temme, A. J. A. M., G. B. M. Heuvelink, J. M. Schoorl, and L. Claessens (2009). "Chapter 5 Geostatistical simulation and error propagation in geomorphometry". In: *Geomorphometry*. Ed. by T. Hengl and H. I. Reuter. Vol. 33. Elsevier, pp. 121–140. DOI: 10.1016/S0166-2481(08)00005-6.

Teufelsbauer, H., Y. Wang, S. P. Pudasaini, R. I. Borja, and W. Wu (2011). "DEM simulation of impact force exerted by granular flow on rigid structures". In: *Acta Geotechnica* 6.3, pp. 119–133. DOI: 10.1007/s11440-011-0140-9.

Uribe, F., I. Papaioannou, W. Betz, and D. Straub (2020). "Bayesian inference of random fields represented with the Karhunen–Loève expansion". In: *Computer Methods in Applied Mechanics and Engineering* 358, p. 112632. DOI: https://doi.org/10.1016/j.cma.2019.112632.

Wallemacq, P. (2018). *Economic losses, poverty & disasters 1998-2017*. Tech. rep. United Nations Office for Disaster Risk Reduction and the Centre for Research on the Epidemiology of Disasters. DOI: 10.13140/RG.2.2.35610.08643.

Walter, F., F. Amann, A. Kos, R. Kenner, M. Phillips, A. de Preux, M. Huss, C. Tognacca, J. Clinton, T. Diehl, and Y. Bonanomi (2020). "Direct observations of a three million cubic meter rock-slope collapse with almost immediate initiation of ensuing debris flows". In: *Geomorphology* 351, p. 106933. DOI: https://doi.org/10.1016/j.geomorph.2019.106933.

Wang, H. Q. and J. L. Li (2018). "Adaptive Gaussian process approximation for Bayesian inference with expensive likelihood functions". In: *Neural Computation* 30.11, pp. 3072–3094. DOI: 10.1162/neco_a_01127.

Watson, C. S., J. Carrivick, and D. Quincey (2015). "An improved method to represent DEM uncertainty in glacial lake outburst flood propagation using stochastic simulations". In: *Journal of Hydrology* 529, pp. 1373–1389. DOI: 10.1016/j.jhydrol.2015.08.046.

Wechsler, S. P. (2007). "Uncertainties associated with digital elevation models for hydrologic applications: a review". In: *Hydrology and Earth system Sciences* 11, pp. 1481–1500. DOI: 10.5194/hess-11-1481-2007.

Wechsler, S. P. and C. N. Kroll (2006). "Quantifying DEM uncertainty and its effect on topographic parameters". In: *Photogrammetric Engineering and Remote Sensing* 72.9, pp. 1081–1090. DOI: 10.14358/PERS.72.9.1081.

Wessel, B., M. Huber, C. Wohlfart, U. Marschalk, D. Kosmann, and A. Roth (2018). "Accuracy assessment of the global TanDEM-X digital elevation model with GPS data". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 139, pp. 171–182. DOI: 10.1016/j.isprsjprs.2018.02.017.

Wiener, N. (1949). *Extrapolation, interpolation and smoothing of stationary time series*. Cambridge, Massachusetts: MIT Press.

Wilson, J. P. (2012). "Digital terrain modeling". In: *Geomorphology* 137, pp. 107–121. DOI: 10.1016/j.geomorph.2011.03.012.

Xia, X. L. and Q. H. Liang (2018). "A new depth-averaged model for flow-like landslides over complex terrains with curvatures and steep slopes". In: *Engineering Geology* 234, pp. 174–191. DOI: 10.1016/j.enggeo.2018.01.011.

Zhang, J. J., W. X. Li, L. Z. Zeng, and L. S. Wu (2016). "An adaptive Gaussian process-based method for efficient Bayesian experimental design in groundwater contaminant source identification problems". In: *Water Resources Research* 52.8, pp. 5971–5984. DOI: https://doi.org/10.1002/2016WR018598.